

UNIVERSIDAD CARLOS III DE MADRID
ESCUELA POLITÉCNICA SUPERIOR
GRADO EN INGENIERÍA INFORMÁTICA

APLICACIÓN WEB PARA LA ELABORACIÓN Y GESTIÓN DE PROCESOS ETL EN BIG DATA

Alumno: Juan Chamorro Rodríguez
Tutor: Harith Aljumaily
23 de Septiembre de 2016

Resumen

Las empresas de hoy en día se desarrollan de manera orgánica y de forma exponencial se van añadiendo más fuentes de datos. Este hecho induce a que empiecen a aparecer nuevos requisitos y sea necesaria una integración de sistemas. Como consecuencia, se ven en la necesidad de disponer de repositorios centralizados estandarizados de todos los datos de la organización y tener una visión global de la totalidad de los datos fijados en un almacén de datos. Esto ofrece la posibilidad a los altos cargos de las empresas de realizar decisiones estratégicas de manera más eficiente y rigurosa. Por estos motivos, a las organizaciones les interesa implantar procesos ETL. El presente trabajo de fin de grado se enmarca dentro de este área concreta del Big Data que se ocupa de mover datos desde múltiples fuentes, reformatearlos y limpiarlos, y cargarlos en otra base de datos o almacén de datos para su posterior análisis.

El cometido de este trabajo es realizar un análisis y desarrollo de una aplicación web que genere procesos ETL de forma centralizada y mediante programación por componentes. La aplicación deberá ser capaz de crear programas autónomos que puedan extraer información desde las principales fuentes de origen, convertir el formato de los datos extraídos para que se ajuste a los requisitos de las fuentes de destino más utilizadas en una organización y escribir los datos en dichas fuentes de destino. Además, deberá incluir un repositorio para el control de versiones de estos procesos, un módulo que permita organizar los procesos en áreas de trabajo, la posibilidad de ejecutar procesos en determinados momentos de forma automática y un módulo que permita al usuario organizar su trabajo. El acceso a la aplicación será restringido, existiendo un sistema de perfiles y usuarios para este fin.

El motivo de estudio y desarrollo de este proyecto viene definido por la necesidad de traer grandes volúmenes de datos desde diversas fuentes en diversos formatos y diferentes niveles de calidad juntos y almacenarlos en un sistema de destino. Para ello es necesario contar con herramientas que permitan realizar estas tareas haciendo frente a los retos que suponen el rendimiento, la facilidad y velocidad de desarrollo y el seguimiento y la capacidad de soporte de la solución. Estas herramientas son conocidas por las siglas ETL (extraer, transformar y cargar).

Índice de contenidos

Resumen.....	1
Índice de contenidos	2
Índice de ilustraciones.....	4
Índice de tablas	6
Glosario de términos	11
1. Introducción	12
1.1. Definición del problema	12
1.2. Objetivos	15
1.3. Propósito del documento.....	16
1.4. Fases del desarrollo.....	17
1.5. Medios empleados	17
1.6. Estructura de la memoria.....	18
2. Estado de la cuestión	19
2.1. Big Data	19
2.2. Procesos ETL.....	24
2.3. Resumen general de las herramientas actuales	39
3. Gestión de proyecto software.....	40
3.1. Alcance del proyecto	40
3.2. Plan de trabajo	41
3.3. Gestión de recursos.....	47
3.4. Gestión de riesgos	49
3.5. Plan de pruebas.....	54
4. Solución	73
4.1. Definición del problema: soluciones	73
4.2. Proceso de desarrollo.....	74
5. Evaluación	145
5.1. Proceso de evaluación.....	145
5.2. Análisis de resultados.....	163
6. Conclusión	167
6.1. Conclusión	167
6.2. Trabajos futuros	168
6.3. Problemas encontrados	169

6.4. Opiniones personales	170
7. Bibliografía	172
Anexo I. Requisitos	177
Anexo II. Control de versiones	215

Índice de ilustraciones

Ilustración 1 - Ejemplo de tablero Kanban	13
Ilustración 2- Diagrama para ilustrar ETL	14
Ilustración 3 - Las 3 Vs de Big Data	21
Ilustración 4 - Comparativa entre Big Data y BI	23
Ilustración 5 - Arquitectura de inteligencia de negocio	24
Ilustración 6 - Elementos del proceso ETL	25
Ilustración 7 - Ejemplo exportación proceso como servicio web.	33
Ilustración 8 - Diagrama de Gantt	46
Ilustración 9 - Proceso de desarrollo por etapas	74
Ilustración 10 - Esquema de los casos de uso del administrador	78
Ilustración 11 - Esquema de los casos de uso del usuario	79
Ilustración 12- Arquitectura de la aplicación	92
Ilustración 13 - Modelo Entidad/Relación de la base de datos de la aplicación	94
Ilustración 14 - Página de acceso	95
Ilustración 15 - Diseño tablero Kanban	96
Ilustración 16 - Diseño añadir actividad de trabajo	97
Ilustración 17 - Diseño selección área de trabajo	98
Ilustración 18 - Diseño incluir trabajo	98
Ilustración 19 - Diseño área de trabajo	99
Ilustración 20 - Diseño área de trabajo 2	100
Ilustración 21 - Diseño área de elaboración de proceso ETL	101
Ilustración 22 - Diseño subida versión	102
Ilustración 23 - Diseño conector componente fichero entrada	103
Ilustración 24 - Diseño mapeo campos componente fichero entrada	104
Ilustración 25 - Diseño ver campos componente fichero entrada	104
Ilustración 26 - Diseño fichero componente fichero entrada	105
Ilustración 27 - Diseño ver campos fichero salida	106
Ilustración 28 - Diseño fichero componente fichero salida	106
Ilustración 29 - Diseño conector componente mapeo	107
Ilustración 30 - Diseño mapeo campos componente mapeo	107
Ilustración 31 - Diseño relaciones componente mapeo	108
Ilustración 32 - Diseño conector componente filtro	109
Ilustración 33 - Diseño mapeo campos componente filtro	109
Ilustración 34 - Diseño filtros componente filtro	110
Ilustración 35 - Diseño programación trabajos	110
Ilustración 36 - Librería requeridas para la implementación	112
Ilustración 37- Estructura de ficheros en Eclipse, parte 1	113
Ilustración 38 - Estructura de ficheros en Eclipse, parte 2	114
Ilustración 39 - Estructura de ficheros en Eclipse, parte 3	115
Ilustración 40 - Clases de configuración	118
Ilustración 41 - clases del Modelo	120

Ilustración 42 - Clases del Modelo parte ETL	121
Ilustración 43 - Clases del Servicio	124
Ilustración 44 - AdminController_parte1	126
Ilustración 45 - AdminController_parte2	126
Ilustración 46 - AreaTrabajoController.....	127
Ilustración 47 - ClaveUsuarioController	128
Ilustración 48 – ControlController.....	128
Ilustración 49 - EtIController_parte1	129
Ilustración 50 - EtIController_parte2	130
Ilustración 51 - MainController	131
Ilustración 52 - Clases dao.....	132
Ilustración 53 - Clases dao etl.....	133

Índice de tablas

Tabla 1 - Tabla de mapa lógico de datos	27
Tabla 2 - Calendario de tareas.....	45
Tabla 3 - Personal encargado del proyecto	48
Tabla 4 - Elementos empleados en el proyecto	48
Tabla 5- Coste total del proyecto	48
Tabla 6 - Coste total del proyecto con beneficios.....	49
Tabla 7 - Formato de tabla de recogida de riesgos	49
Tabla 8 - Tabla de riesgos	54
Tabla 9 - Formato de pruebas	55
Tabla 10 - Pruebas de planificación de actividades de trabajo	57
Tabla 11 - Pruebas de gestión y elaboración de trabajos	64
Tabla 12- Pruebas de programación y ejecución de trabajos	66
Tabla 13 - Pruebas de administración	71
Tabla 14 - Formato tabla descripción de requisitos	76
Tabla 15 - Formato de las tablas de especificación de casos de uso	79
Tabla 16 - Caso de uso CU-1	80
Tabla 17 - Caso de uso CU-2.....	81
Tabla 18 - Caso de uso CU-3	82
Tabla 19 - Caso de uso CU-4.....	82
Tabla 20 - Caso de uso CU-5	86
Tabla 21 - Caso de uso CU-6.....	87
Tabla 22 - Caso de uso CU-7	87
Tabla 23 - Matriz de trazabilidad de los Requisitos de SW Funcionales y los Casos de Uso.....	90
Tabla 24 - Relación enlace-plantilla-controlador	135
Tabla 25 – Pruebas finales de planificación de actividades de trabajo.....	142
Tabla 26 – Pruebas finales de gestión y elaboración de trabajos	142
Tabla 27- Pruebas finales de programación y ejecución de trabajos.....	143
Tabla 28 - Pruebas finales de administración	144
Tabla 29 - Formato de tabla de evaluación de escenarios.....	146
Tabla 30 - Escenario E-1	151
Tabla 31 - Escenario E-2	152
Tabla 32 - Escenario E-3	154
Tabla 33 - Escenario E-4	160
Tabla 34 - Escenario E-5	161
Tabla 35 - Escenario E-6	163
Tabla 36 - Matriz de trazabilidad de requisitos vs escenarios de evaluación	166
Tabla 37 - Tabla de Bibliografía	176
Tabla 38 - Requisito RRG-01.....	177
Tabla 39 - Requisito RRG-02.....	177
Tabla 40 - Requisito RRG-03.....	177
Tabla 41 - Requisito RFAA-04	178

Tabla 42 - Requisito RRAA-05.....	178
Tabla 43 - Requisito RRAA-06.....	178
Tabla 44 - Requisito RRAA-07.....	178
Tabla 45 - Requisito RRAA-08.....	179
Tabla 46 - Requisito RRAA-09.....	179
Tabla 47 - Requisito RRAA-10.....	179
Tabla 48 - Requisito RFAA-11.....	179
Tabla 49 - Requisito RRAA-12.....	180
Tabla 50 - Requisito RFAA-13.....	180
Tabla 51 - Requisito RRAA-14.....	180
Tabla 52 - Requisito RRAA-15.....	180
Tabla 53 - Requisito RFAA-16.....	181
Tabla 54 - Requisito RRAA-17.....	181
Tabla 55 - Requisito RRAA-18.....	181
Tabla 56 - Requisito RFAA-19.....	181
Tabla 57 - Requisito RRAA-20.....	182
Tabla 58 - Requisito RRAA-21.....	182
Tabla 59 - Requisito RRAA-22.....	182
Tabla 60 - Requisito RRAA-23.....	182
Tabla 61 - Requisito RFAA-24.....	183
Tabla 62 - Requisito RRAA-25.....	183
Tabla 63 - Requisito RFAA-26.....	183
Tabla 64 - Requisito RRAA-27.....	183
Tabla 65 - Requisito RRAA-28.....	184
Tabla 66 - Requisito RFAU-29.....	184
Tabla 67 - Requisito RRAU-30.....	184
Tabla 68 - Requisito RRAU-31.....	184
Tabla 69 - Requisito RRAU-32.....	185
Tabla 70 - Requisito RRAU-33.....	185
Tabla 71 - Requisito RRAU-34.....	185
Tabla 72 - Requisito RRAU-35.....	185
Tabla 73 - Requisito RRAU-36.....	186
Tabla 74 - Requisito RFAU-37.....	186
Tabla 75 - Requisito RRAU-38.....	186
Tabla 76 - Requisito RRAU-39.....	186
Tabla 77 - Requisito RRAU-40.....	187
Tabla 78 - Requisito RRAU-41.....	187
Tabla 79 - Requisito RFAU-42.....	187
Tabla 80 - Requisito RRAU-43.....	187
Tabla 81 - Requisito RRAU-44.....	188
Tabla 82 - Requisito RFAU-45.....	188
Tabla 83 - Requisito RRAU-46.....	188
Tabla 84 - Requisito RFAU-47.....	188
Tabla 85 - Requisito RRAU-48.....	189

Tabla 86 - Requisito RRAU-49	189
Tabla 87 - Requisito RRAU-50	189
Tabla 88 - Requisito RRAU-51	189
Tabla 89 - Requisito RFAU-52	190
Tabla 90 - Requisito RRAU-53	190
Tabla 91 - Requisito RFAU-54	190
Tabla 92 - Requisito RRAU-55	190
Tabla 93 - Requisito RRAU-56	191
Tabla 94 - Requisito RFTP-57	191
Tabla 95 - Requisito RRTP-58	191
Tabla 96 - Requisito RRTP-59	191
Tabla 97 - Requisito RRTP-60	192
Tabla 98 - Requisito RFTP-61	192
Tabla 99 - Requisito RRTP-62	192
Tabla 100 - Requisito RFTP-63	192
Tabla 101 - Requisito RRTP-64	193
Tabla 102 - Requisito RFTP-65	193
Tabla 103 - Requisito RRTP-66	193
Tabla 104 - Requisito RFTP-67	193
Tabla 105 - Requisito RRTP-68	194
Tabla 106 - Requisito RFTP-69	194
Tabla 107 - Requisito RRTP-70	194
Tabla 108 - Requisito RFTE-71	195
Tabla 109 - Requisito RRTE-72	195
Tabla 110 - Requisito RFTE-73	195
Tabla 111 - Requisito RRTE-74	195
Tabla 112 - Requisito RRTE-75	196
Tabla 113 - Requisito RFTE-76	196
Tabla 114 - Requisitos RRTE-77	196
Tabla 115 - Requisito RFTE-78	196
Tabla 116 - Requisito RRTE-79	197
Tabla 117 - Requisito RRTE-80	197
Tabla 118 - Requisito RFTE-81 [ELIMINADO]	197
Tabla 119 - Requisito RRTE-82 [ELIMINADO]	197
Tabla 120 - Requisito RRTE-83 [ELIMINADO]	198
Tabla 121 - Requisito RFTG-84	198
Tabla 122 - Requisito RRTG-85	198
Tabla 123 - Requisito RRTG-86	198
Tabla 124 - Requisito RRTG-87	199
Tabla 125 - Requisito RFTG-88	199
Tabla 126 - Requisito RFTG-89	199
Tabla 127 - Requisito RRTG-90	199
Tabla 128 - Requisito RFTG-91	200
Tabla 129 - Requisito RRTG-92	200

Tabla 130 - Requisito RRTG-93.....	200
Tabla 131 - Requisito RRTG-94.....	200
Tabla 132 - Requisito RRTG-95.....	201
Tabla 133 - Requisito RFTG-96.....	201
Tabla 134 - Requisito RRTG-97.....	201
Tabla 135 - Requisito RFTG-98.....	201
Tabla 136 - Requisito RRTG-99.....	202
Tabla 137 - Requisito RRTG-100.....	202
Tabla 138 - Requisito RFTG-101.....	202
Tabla 139 - Requisito RRTG-102.....	202
Tabla 140 - Requisito RFTG-103.....	203
Tabla 141 - Requisito RRTG-104.....	203
Tabla 142 - Requisito RFTG-105.....	203
Tabla 143 - Requisito RRTG-106.....	203
Tabla 144 - Requisito RFTG-107.....	204
Tabla 145 - Requisito RRTG-108.....	204
Tabla 146 - Requisito RFTG-109.....	204
Tabla 147 - Requisito RRTG-110.....	204
Tabla 148 - Requisito RFTG-111.....	205
Tabla 149 - Requisito RRTG-112.....	205
Tabla 150 - Requisito RRTG-113.....	205
Tabla 151 - Requisito RRTG-114.....	205
Tabla 152 - Requisito RRTG-115.....	206
Tabla 153 - Requisito RRTG-116.....	206
Tabla 154 - Requisito RRTG-117.....	206
Tabla 155 - Requisito RRTG-118.....	206
Tabla 156 - Requisito RRTG-119.....	207
Tabla 157 - Requisito RRTG-120.....	207
Tabla 158 - Requisito RRTG-121.....	207
Tabla 159 - Requisito RRTG-122.....	207
Tabla 160 - Requisito RRTG-123.....	208
Tabla 161 - Requisito RRTG-124.....	208
Tabla 162 - Requisito RRTG-125.....	208
Tabla 163 - Requisito RRTG-126.....	208
Tabla 164 - Requisito RRTG-127.....	209
Tabla 165 - Requisito RRTG-128.....	209
Tabla 166 - Requisito RRTG-129.....	209
Tabla 167 - Requisito RRTG-130.....	209
Tabla 168 - Requisito RRTG-131.....	210
Tabla 169 - Requisito RRTG-132.....	210
Tabla 170 - Requisito RRTG-133.....	210
Tabla 171 - Requisito RRTG-134.....	210
Tabla 172 - Requisito RRTG-135.....	211
Tabla 173 - Requisito RRTG-136.....	211

Tabla 174 - Requisito RRTG-137	211
Tabla 175 - Requisito RRTG-138.....	211
Tabla 176 - Requisito RRTG-139.....	212
Tabla 177 - Requisito RRTG-140.....	212
Tabla 178 - Requisito RRTG-141.....	212
Tabla 179 - Requisito RRTG-142.....	213
Tabla 180 - Requisito RRTG-143.....	213
Tabla 181 - Requisito RRTG-144.....	213
Tabla 182 - Requisito RRTG-145.....	213
Tabla 183 - Requisito RRTG-146.....	214
Tabla 184 - Requisito RRTG-147.....	214
Tabla 185 - Requisito RRTG-148.....	214
Tabla 186 - Requisito RRTG-149.....	214
Tabla 187 - Control de versiones.....	215

Glosario de términos

Catálogo de términos específicos del contexto de trabajo.

Análisis OLAP	Es una solución utilizada en Business Intelligence cuyo objetivo es agilizar la consulta de grandes cantidades de datos. Para ello utiliza estructuras multidimensionales que contienen datos resumidos de grandes bases de datos o sistemas transaccionales.
Big Data:	Engloba infraestructuras, tecnologías y servicios que han sido creados para dar solución al procesamiento de enormes conjuntos de datos estructurados, semi-estructurados o no estructurados que pueden provenir de distintas fuentes.
Business Intelligence	Transformar los datos en información, y la información en conocimiento, de manera que se pueda optimizar el proceso de toma de decisiones en los negocios.
Dashboard	Es una herramienta de visualización de datos que presenta el estado actual de las métricas e indicadores claves del rendimiento de una empresa o área de negocio.
Data mining	Se centra en encontrar gracias al reconocimiento de patrones de datos información oculta que se encuentra implícitamente en los datos.
ETL:	Proceso que organiza el flujo de los datos entre diferentes sistemas en una organización y aporta los métodos y herramientas necesarios para mover datos desde múltiples fuentes a un almacén de datos, reformatearlos, limpiarlos y cargarlos en otra base de datos, data mart ó almacén de datos.
Framework	Es un esquema para el desarrollo y/o la implementación de aplicaciones.
Hibernate	Es una herramienta de Mapeo objeto-relacional (ORM) para la plataforma Java que facilita el mapeo de atributos entre una base de datos relacional tradicional y el modelo de objetos de una aplicación, mediante archivos declarativos (XML) o anotaciones en los beans de las entidades que permiten establecer estas relaciones.
Lookup	Devuelve los valores de una tabla de base de datos o un archivo plano asociado con un determinado valor de entrada. En SQL podría ser considerado como una sub consulta.
Mapeo	En el contexto de este proyecto, se entiende como la fragmentación de la información de una fuente de datos en unidades que representan ideas esenciales, pero con la propiedad de poder establecer vínculos entre las mismas. Al mismo tiempo, estas unidades pueden incorporar otro tipo de informaciones.
Kanban:	El sistema Kanban para el Software es un enfoque sin iteraciones para organizar el trabajo.
Trabajo:	En el contexto de este proyecto, sinónimo de proceso ETL.

1. Introducción

El objeto de este capítulo es que el lector tenga conocimiento de las razones por las que el autor del trabajo ha elegido el tema planteado, es decir, el sentido y finalidad de este proyecto. Con este fin, se presenta el problema que ha propiciado el trabajo con una exposición del conocimiento existente en torno al problema, los objetivos marcados, la línea de trabajo seguida y la descripción de los capítulos detallando la metodología empleada.

1.1. Definición del problema

La idea principal de este proyecto surge a raíz de los puntos débiles que se han detectado en las herramientas ETL que existen actualmente en el mercado. Los principales y que han llevado a realizar la aplicación que se plantea en este proyecto son la necesidad de instalar y actualizar la herramienta en cada cliente, requerimientos especiales de software y librerías, están hechas para un sistema operativo específico y su portabilidad es escasa. Estas desventajas son propias de aplicaciones de escritorio, por lo que se pretende diseñar e implementar una herramienta ETL como una aplicación web para poder solventarlas.

Existe otro problema con las herramientas de software ETL actuales que se ha tenido en cuenta. Estas herramientas han ido evolucionando de tal manera que incluyen funcionalidades avanzadas de Business Intelligence como por ejemplo análisis OLAP, reportes, dashboard o data mining, dando lugar a herramientas poco intuitivas y difíciles de entender. Por este motivo, la aplicación que se presenta ofrecerá exclusivamente la funcionalidad tradicional de un proceso ETL de Big Data.

Hasta el momento, se ha centrado el objetivo en intentar mejorar las propias herramientas ETL, pero también es importante estudiar el grupo de trabajo que hace uso de las mismas. A este respecto, cada vez es más común encontrar equipos especializados en producir procesos ETL dentro de los departamentos de desarrollo de las empresas y, como en cualquier otro equipo de trabajo, nace el problema de cómo optimizar las tareas y el de cómo acotar responsabilidades. Para solucionar el primer problema, es decir, para ayudar a gestionar mejor el trabajo, se ofrece un módulo que incluye un tablero Kanban similar al siguiente:

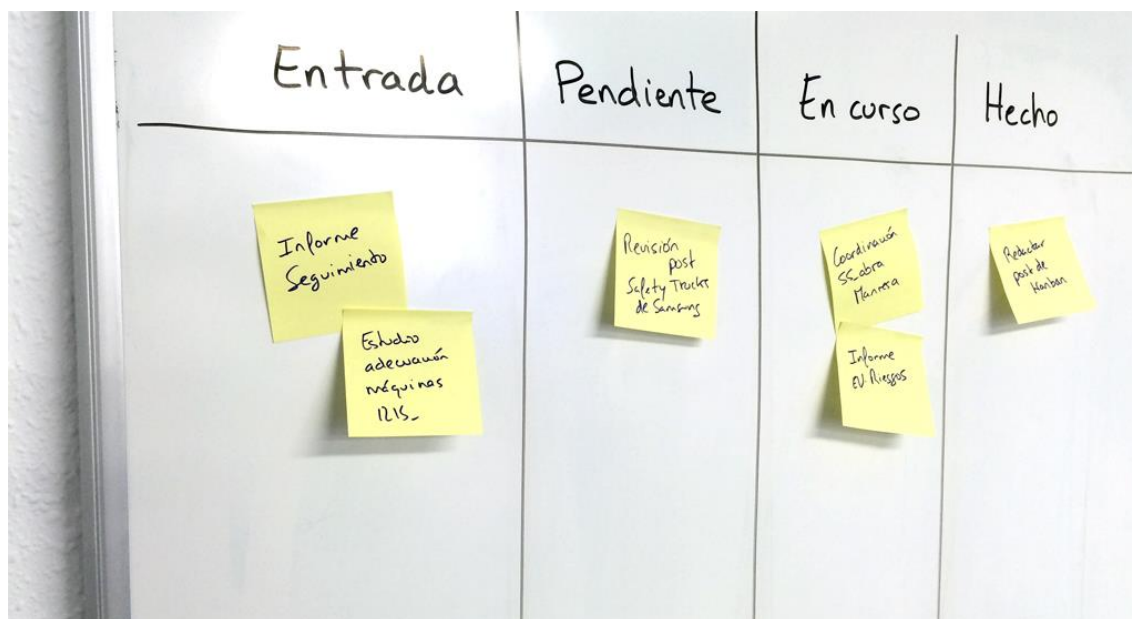


Ilustración 1 - Ejemplo de tablero Kanban

La primera columna del tablero que se muestra en la ilustración número 1 se corresponde con la bandeja de entrada, donde van llegando las tareas. La segunda hace referencia a las tareas que se ha determinado que se van a hacer. Seguidamente, la tercera columna indica las tareas que se están llevando a cabo. Por último, la cuarta columna refleja las tareas terminadas. Es una forma muy sencilla y visual de poder analizar si la carga de trabajo es la adecuada y de conocer el estado de las tareas. Cabe señalar que Kanban es un modelo inventado en los años 40 por la empresa automovilística Toyota para gestionar cadenas de producción y que en la actualidad se está empleando para gestionar proyectos, en este caso, se ha adaptado al mundo de Big Data, donde las tareas serán procesos ETL.

Para solucionar el segundo problema, es decir, para acotar responsabilidades y compartimentar el trabajo, se facilita un sistema de áreas de trabajo dentro de los cuales irán los procesos ETL. Para que un usuario pueda acceder a los procesos de un área de trabajo determinado deber estar asociado a éste. En las herramientas ETL más conocidas se pueden organizar los procesos en directorios, pero cualquier usuario con acceso a la misma puede acceder a cualquier proceso de cualquier directorio.

Con el fin de comprender el contexto de trabajo es necesario exponer de manera concisa la tecnología Big Data en general y los procesos ETL en particular. Por Big Data se entiende la captura y el manejo de gran cantidad de información. Según un gran número de estudios de investigación y de mercado independientes las cantidades de datos se multiplican por dos cada año. Además, un gran número de compañías almacenan los datos históricos de dos o más años. Big Data se encarga de explotar esos volúmenes de información y de nuevos tipos de datos con nuevas clases de herramientas. Muchas de las aplicaciones que se utilizaban para hacer frente a una cantidad de información relativamente grande no pueden sostenerse hoy en día con Big Data. Esa nueva generación de herramientas posee el potencial para transformar la forma en que funciona el negocio de las organizaciones.

Dentro de esta nueva generación de herramientas de Big Data se encuentran las denominadas ETL, software que se ocupa de la integración de Big Data, en particular, de la extracción de datos

tanto homogénea como heterogénea, transformar los datos para su almacenamiento en el formato adecuado o estructura para las consultas y análisis y, además, cargar estos datos en base de datos o en el sistema de destino que corresponda.

La ilustración número 2 que se muestra seguidamente representa el flujo de información desde que se obtiene a través de distintas fuentes de entrada hasta la generación de valor para la organización. Esta representación permite visualizar el papel que ocupa un proceso ETL dentro de este flujo.

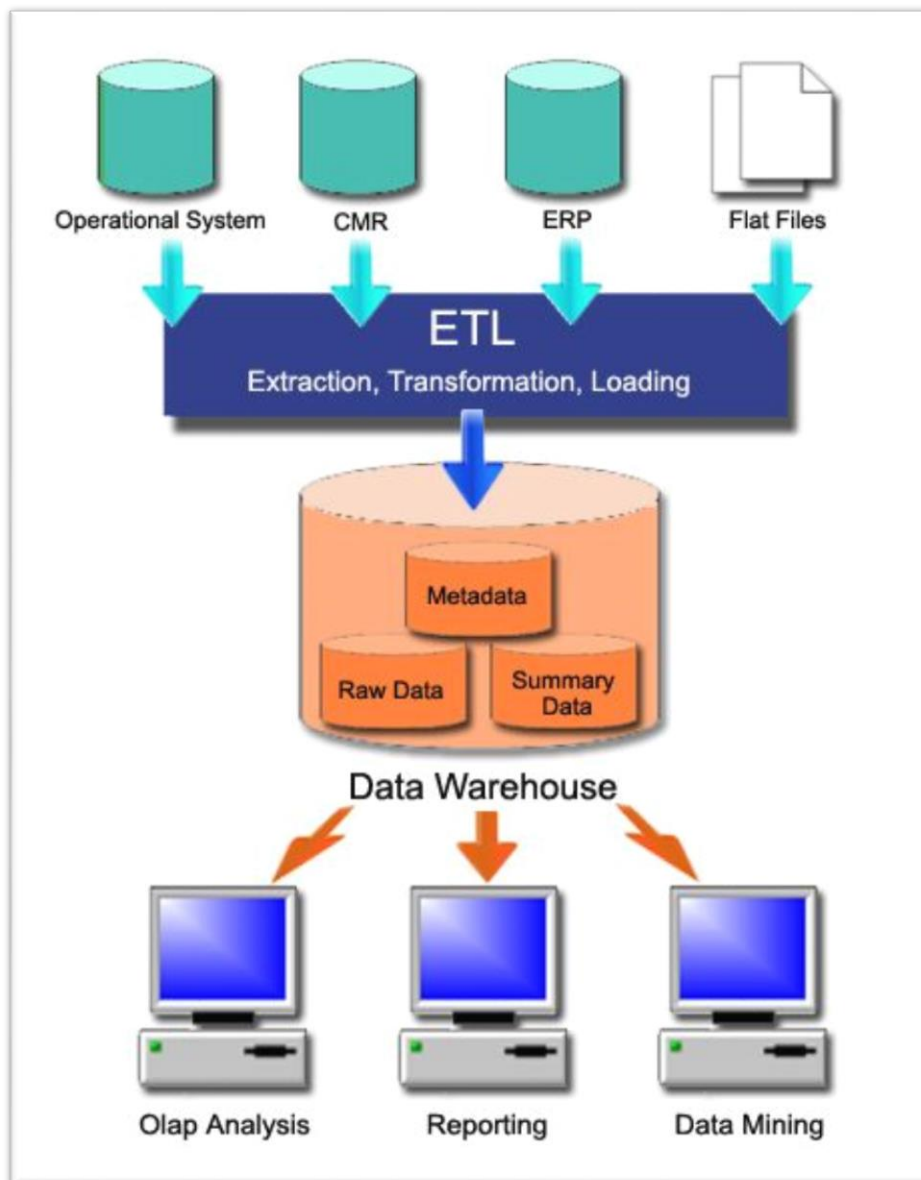


Ilustración 2- Diagrama para ilustrar ETL

Existen una serie de retos a los que se tienen que enfrentar las herramientas ETL. Un sistema ETL que no esté bien diseñado puede producir graves problemas operativos. Asimismo, debe tener la habilidad de conectar con un nutrido rango de tipos de estructuras de datos, que integran, entre otros, bases de datos (relacionales y no relacionales), variados formatos de ficheros y repositorios de contenido. De la misma forma, debe tener la habilidad de transformar los datos, desde transformaciones básicas (conversiones de tipos, manipulación de cadenas o

cálculos simples), transformaciones intermedias (agregaciones, sumalizaciones, lookups) hasta transformaciones complejas como análisis de texto en formato libre o texto enriquecido.

La aplicación que se ha gestado en este proyecto soporta algunos de los tipos de estructuras de datos más empleados, así como algunas de las funcionalidades más habituales para poder realizar las transformaciones de los datos con los que haya que tratar. Se ha concebido como una herramienta que pueda dar pie a realizar futuros proyectos que se ocupen de enriquecer la misma añadiendo más componentes ETL. A modo de ejemplo, se podrían añadir componentes de entrada y de salida para poder trabajar con la base de datos relacional PostgreSQL, ya que no se incluyen en la presente versión.

Siguiendo con los retos a los que se enfrentan las herramientas ETL, se encuentra el de adaptarse a las distintas plataformas hardware y sistemas operativos existentes. Como ya se vio anteriormente, este problema se solventa con aplicaciones web.

En resumen, todos los puntos débiles que se han encontrado en las herramientas ETL con mayor demanda, la importancia que representan los procesos ETL en Big Data y éste a su vez dentro de las compañías presentes en la actualidad y los equipos de trabajo que han surgido en torno a este nuevo tipo de trabajo han dado lugar al diseño y a la implementación de la aplicación web para la elaboración y gestión de procesos ETL que se ha llevado a cabo en este trabajo de fin de grado.

1.2. Objetivos

En las siguientes líneas se explican los objetivos del presente trabajo fin de grado para su comprensión. Se pretende ofrecer un punto de vista antes de abordar el tratamiento de toda la información producida en los capítulos venideros, por lo que se trata de un anticipo de los mismos.

- Estudiar el ámbito de las tecnologías en las que se desarrolla el trabajo, es decir, Big Data haciendo hincapié en la extracción, transformación y almacenamiento de datos, detallando cada una de estas partes.
- Estudiar y analizar las herramientas ETL vigentes en el mercado para así conocer los nuevos aportes que puede ofrecer la nueva aplicación a realizar en relación a las otras presentes en el mercado.
- Diseñar e implementar una aplicación que incluya las siguientes funcionalidades:
 - Facilitar la generación de procesos ETL mediante una aplicación web instalada en un servidor y que puede ser utilizada desde cualquier equipo con acceso a la red por usuarios con credenciales. Ofrece una interfaz donde diseñar los procesos de forma sencilla e intuitiva que consiste en ir uniendo distintos componentes con funcionalidades diversas mediante sus flujos de entrada y flujos de salida para realizar una tarea más compleja.
 - Ofrecer un administrador regular de procesos en segundo plano que ejecuta procesos a intervalos regulares (por ejemplo, cada minuto, día, semana o mes). Los procesos que deben ejecutarse y la hora en la que deben hacerlo se

especifican desde la interfaz de la aplicación. Sólo se admitirán procesos ETL generados desde la aplicación. Además, los procesos programados desde esta interfaz se ejecutarán desde el propio servidor de la aplicación, por lo que éste debe contar con acceso a las fuentes requeridas por estos procesos para poder ejecutarlos.

- Proporcionar un control de versiones de los procesos realizados.
 - Clasificar los procesos y restringir el acceso a los mismos mediante áreas de trabajo. Es decir, se crean áreas de trabajo que emulan directorios físicos, dentro de estas áreas de trabajo se construyen los procesos ETL. El acceso a cada área de trabajo es restringido, por lo que un usuario tiene que tener asociado un área de trabajo para poder trabajar con los procesos que hay en su interior.
 - Descargar un proceso generado desde la aplicación en el equipo del usuario con todas las librerías requeridas para que pueda ser ejecutado.
 - Sistema de tarjetas basado en Kanban que permita a los usuarios gestionar mejor su trabajo.
- Efectuar pruebas de evaluación cuyo propósito es demostrar que los objetivos a desarrollar se han cumplido y que la aplicación efectúa todas las funcionalidades que se han detallado en los puntos indicados anteriormente.

1.3. Propósito del documento

Mediante la presente memoria se pretende mostrar al lector el trabajo desempeñado durante toda la vida del TFG, indicando el recorrido que se ha seguido a través de las distintas fases desarrolladas.

Entre estas fases se encuentran aquella en la que se congrega todos los requisitos que tiene que cumplir el sistema, aquella en la que se realiza un esbozo de lo requerido, documentando la interfaz y el manejo de datos del software, aquella en la que se implementa el diseño del software en el lenguaje de programación y con los frameworks detallados en el propio diseño y aquella en la que se determina que no se hayan generado errores en etapas precedentes.

1.4. Fases del desarrollo

- Estado del arte

En esta etapa se analiza el conocimiento que ya se obtuvo respecto del problema planteado. Por consiguiente, en esta fase se estudiarán las herramientas más destacadas que pone a disposición el mercado sobre ETL en Big Data y las tecnologías a aplicar para poder determinar y escoger aquello que resulte más conveniente a la hora de encarar el proyecto con éxito.

- Definición del problema

En esta etapa se estudia cómo trabajan las distintas herramientas ETL del mercado en entornos Big Data y cómo se podrían mejorar. Con este fin, se explicará qué es Big Data y el cometido de los procesos ETL dentro de éste.

- Solución al problema

En esta fase se detallan todos los pasos seguidos en la solución que se ha realizado, incluida la batería de pruebas llevada a cabo tras la conclusión de la implementación del código.

- Evaluación

Se busca conocer qué tanto el proyecto ha conseguido satisfacer sus objetivos. Se presentarán los procesos llevados a cabo para determinar si la aplicación realizada funciona según se espera. Para poder realizar esta evaluación se seleccionarán las distintas funcionalidades de la aplicación en primer lugar y, en el siguiente paso, se procederá a analizarlas.

1.5. Medios empleados

Por una parte, se tienen aquellas fuentes bibliográficas que han sido empleadas para poder encontrar una solución solvente al problema planteado:

- Libros en formato PDF y DOCX.
- Recursos tecnológicos que ofrece la Biblioteca Digital de la UC3M.
- Páginas Web oficiales de aplicaciones que resuelven problemas similares a los planteados.
- Herramientas con licencia libre que resuelven problemas similares a los planteados.
- Blogs con contenido tecnológico.
- Conocimientos extraídos del trabajo realizado en el departamento de desarrollo de uno de los centros de la compañía Konecta BTO S.L., a la que actualmente pertenece el alumno autor de este proyecto. Esta compañía se dedica a la externalización de procesos de negocio.

Por otra parte, se tienen aquellos frameworks y librerías empleados para facilitar la implementación de la aplicación, así como el entorno de trabajo en el que se desenvuelve la misma:

- **Debian 8 (<<jessie>>)**: como sistema operativo en el cual se instalan las herramientas necesarias para el desarrollo y funcionamiento de la aplicación.

- **Eclipse Java EE IDE (<<Mars>>):** herramientas para desarrolladores de Java que crean Java EE y aplicaciones web, incluyendo un IDE Java, herramientas para Java EE, JPA, JSF, Mylyn, EGit y otros.
- **Apache Tomcat 8:** funciona como un contenedor de servlets donde se desplegará la aplicación web.
- **Maven:** es una herramienta de gestión de proyectos que va a permitir manejar las dependencias del proyecto.
- **Spring Framework 4:** es un framework para el desarrollo de aplicaciones y contenedor de inversión de control, de código abierto para la plataforma Java.
- **AngularJS:** es un framework de JavaScript de código abierto que se utiliza para crear y mantener aplicaciones web de una sola página. Su objetivo es aumentar las aplicaciones basadas en navegador con capacidad de Modelo Vista Controlador (MVC).
- **Twitter Bootstrap:** es un framework para el diseño de aplicaciones web. Contiene plantillas de diseño con tipografía, formularios, botones, menús de navegación y otros elementos de diseño basado en HTML y CSS, así como extensiones de JavaScript opcionales adicionales.
- **Jsplumb:** librería gráfica de JavaScript para dibujar flujogramas. Permite representar componentes ETL y las relaciones entre ellos de forma gráfica e intuitiva.
- **Quartz (scheduler):** es una biblioteca de programación de trabajos que se pueden integrar en una amplia variedad de aplicaciones Java. Empleado para programar los procesos ETL desde el propio servidor de la herramienta.

1.6. Estructura de la memoria

Esta memoria se compone de los siguientes capítulos:

- **Introducción al trabajo elaborado y contenido de la memoria**
- **Estado de la cuestión:** se determinarán y evaluarán las distintas líneas de investigación que existan acerca del tema planteado. Es decir, se estudiarán las principales herramientas ETL que existan en el presente para poder solventar el problema a encarar.
- **Gestión del proyecto software:** se encargará de mostrar hasta donde llega el proyecto, el plan de trabajo seguido, cómo se manejan los recursos requeridos para poder encarar este proyecto, los riesgos que puedan acontecer y las pruebas planteadas.
- **Solución:** se detallarán todos los pasos seguidos en la solución realizada, incluida la batería de pruebas llevada a cabo tras la conclusión de la implementación del código.
- **Evaluación:** se presentarán los procesos llevados a cabo para determinar si la aplicación realizada funciona según se espera. Para poder realizar esta evaluación se seleccionarán las distintas funcionalidades de la aplicación en primer lugar y, acto seguido, se procederá a analizarlas.
- **Conclusión:** se recapitularán los puntos fundamentales del trabajo haciendo una reflexión ordenada sobre los aspectos del tema tratado.
- **Bibliografía:** los libros, artículos y materiales empleados para la elaboración del TFG serán recogidos en este apartado.
- **Anexos:** ofrecen instrumentos que completarán la información del proyecto.

2. Estado de la cuestión

En este capítulo se detalla el contexto de trabajo que engloba a este proyecto. Por consiguiente, se explicará qué es Big Data y el papel que juega en el rendimiento de una organización. Se hará especial hincapié en explicar el proceso que se encarga del movimiento y transformación de datos, puesto que el objetivo principal de este proyecto es estudiar las herramientas ETL que están destinadas a realizar esta labor y, al mismo tiempo, ofrecer una alternativa solvente que cubra las necesidades encontradas. Este punto cubrirá también el estudio de estas herramientas, centrándose en las más representativas.

2.1. Big Data

La importancia de Big Data no se basa en la cantidad de datos que se tenga, sino lo que se haga con estos. Lo positivo de esta tecnología es que permite conseguir los datos desde cualquier fuente y estudiarlos para poder obtener respuestas que produzcan una disminución tanto de tiempo como de costes, desarrollo de nuevos productos y, no menos importante, la toma de decisiones denominadas inteligentes que en muchas ocasiones marcan el devenir de una compañía. En el momento en que se engrana Big Data con análisis de alta potencia se pueden desarrollar actividades relacionadas con el negocio, entre otras las que se indican a continuación:

- Dirimir los motivos de los problemas, los fracasos y defectos en tiempo prácticamente real, evitando gastar una suma considerable de dinero para dar con ellos.
- Descubrir comportamientos fraudulentos antes de que desestabilicen el buen funcionamiento de la empresa.
- Simplificar los procesos actuales y controlar el negocio, consiguiendo de esta manera reducir costes.
- Mejorar la visibilidad del negocio a través de la elaboración de informes más precisos.
- Realizar sistemas predictivos que permitan vaticinar posibles complicaciones que puedan acontecer en un futuro.
- Procurar información disponible en todo momento sin necesidad de esperar actualizaciones mediante cuadros de mandos en tiempo real.

Estas tareas que se han citado anteriormente pueden parecer genéricas, para poder ponerlas en situación se van a contextualizar de forma breve dentro de un área de trabajo de una consultora existente dedicada a la externalización de procesos de negocio de un banco:

En el departamento de Venta de Carteras los analistas se encargan de complementar la información de cuentas (entre los que se encuentran los datos propios de la cuenta, los datos judiciales y el expediente físico), para que posteriormente puedan ser vendidas a otros bancos. Parte de los datos de las cuentas con los que se inician las gestiones proceden de ficheros planos cargados automáticamente bajo demanda y otra parte de bases de datos relacionales que son alimentadas por otras unidades como la de Contencioso. Mediante informes y cuadros de mando los responsables del banco pueden conocer en tiempo real las operaciones finalizadas por cada analista, el tiempo que han tardado en realizar cada trámite, si se han cometido irregularidades y demás información relevante. Gracias a esta información que es fruto de la combinación de Big Data y análisis de alta potencia se pueden realizar las actividades

relacionadas con el negocio de las que anteriormente se ha hablado. Como se ha podido apreciar en este caso real, los sistemas que se ocupan de manipular grandes conjuntos de datos adquieren un papel muy relevante dentro del negocio.

2.1.1. *Las 5 magnitudes claves de Big Data*

Big Data no es otra cosa que los sistemas de información que surgieron como respuesta ante las limitaciones que sufrían las tecnologías tradicionales basadas esencialmente en bases de datos relacionales. Existen fundamentalmente tres características de la información que provocaron la aparición de esta nueva tecnología: Volumen, Velocidad y Variedad. Esta manera tridimensional de plasmar los retos del crecimiento de datos y oportunidades fue acuñada en un informe de investigación [22] y conferencias relacionadas por Doug Laney, analista de la compañía META Group. A continuación, se van a explicar en detalle:

- **Volumen:** se refiere a la cantidad de datos a tratar. El tamaño de los datos disponibles ha estado creciendo a un ritmo creciente. Más fuentes de datos son añadidas de forma continua. Antiguamente, todos los datos de las empresas eran generados internamente por los propios empleados. Hoy en día, además de los empleados, los clientes y las máquinas producen estos datos. Más fuentes de datos con un volumen de datos muy superior a lo que venía siendo habitual necesitan ser analizado. Este aspecto es crucial para aquellos que piensen en utilizar esos datos en lugar de permitir que se diluyan con el tiempo. A medida que crecen las bases de datos de las aplicaciones la arquitectura diseñada para soportar los datos necesita ser reevaluada con bastante frecuencia. En la actualidad, es habitual disponer de Terabytes y Petabytes en los sistemas de almacenamiento de las compañías. En el curso 'Industrialización de la inteligencia' impartido en la Universidad Internacional Menéndez Pelayo en el año 2013, el responsable mundial de big-data analytics de Accenture y el responsable de la cátedra de Inteligencia Analítica Avanzada de la Universidad de Oviedo destacaron el hecho de que la información ya se estaba multiplicando "de una manera brutal: en 2020 el volumen de datos almacenados en el mundo será 44 veces mayor que el de 2010". Por aquella fecha, según los datos que manejaban en la cátedra de la Universidad de Oviedo, "sólo el 4,8% de las empresas (españolas) han integrado la tecnología Big Data" [23].
- **Velocidad:** se refiere no solo a la alta frecuencia con la que se generan nuevos datos, sino a la necesidad de dar respuesta a la información en tiempo real, es decir, la velocidad a la que los datos son creados, almacenados, analizados y visualizados. Antiguamente, cuando el procesamiento por lotes estaba a la orden del día, era habitual recibir actualizaciones diarias o incluso semanales. El problema de esto es que los servidores necesitan un tiempo considerable para tratar los datos y actualizar las bases de datos y, en estos tiempos en los que se está inmerso bajo la gran ola de los datos en la que éstos se crean en tiempo real o casi en tiempo real y la ventana de actualización se ha reducido a fracciones de segundos en muchos casos, son procedimientos válidos pero al mismo tiempo insuficientes.

- **Variedad:** se refiere a la naturaleza diversa de la información a manejar. Se viene de trabajar con datos estructurados que encajaban perfectamente en el modelo relacional, pero ahora además se tienen entre manos datos semi-estructurados, no estructurados y aquellos estructurados más complejos que requieren de nuevos métodos de persistencia y consulta. Dicho de otra manera, la extensa variedad de datos exige un enfoque distinto, así como diferentes técnicas para almacenar todos los datos en bruto. Existe una gran variedad de tipos de datos y cada uno de los mismos necesita distintos tipos de análisis y herramientas específicas para poder tratarlos. Entre estos formatos en los que pueden venir almacenados los datos se encuentran csv, pdf, txt, xml, yaml y un largo etcétera.

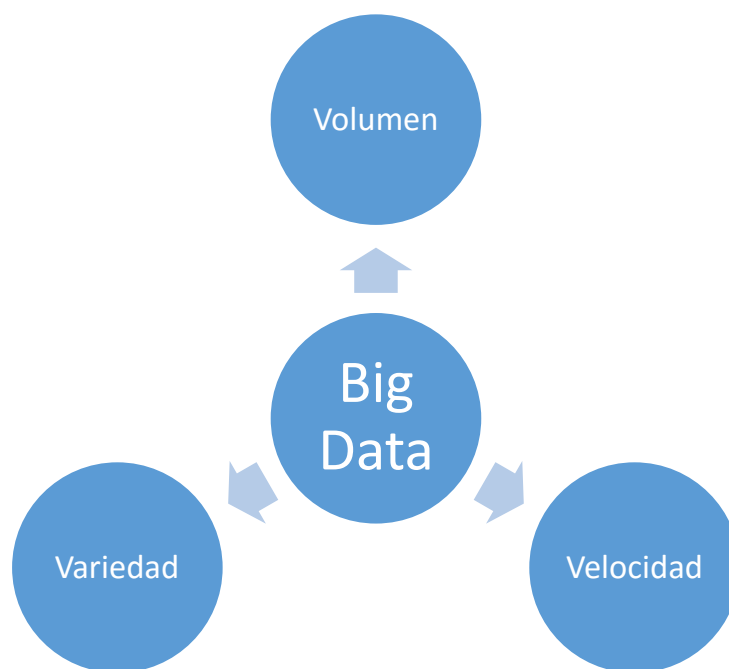


Ilustración 3 - Las 3 Vs de Big Data

Por otro lado, existen dos Vs cada vez más consolidadas a parte de las tradicionales que se han visto antes. Estas Vs es importante que sean también tenidas en cuenta por las organizaciones a la hora de desarrollar una estrategia de Big Data:

- **Veracidad:** el problema de manejar una cantidad de datos tan grande es que la calidad y precisión son menos controlables. Por ende, es fundamental poder medir y evaluar la veracidad de los datos de los que se dispongan por medio de Big Data, para de esta forma poder sacarles el máximo partido posible. Es importante tener en cuenta que no sirve de nada poder gestionar volúmenes de datos descomunales a gran velocidad si esos datos son incorrectos. Dicho lo cual, las organizaciones deben asegurarse de que tanto los datos como el análisis posterior sobre éstos son

correctos, sobre todo en la toma de decisiones automatizada al no existir intervención humana.

- **Valor:** es fundamental producir valor de negocio con los datos que facilita la tecnología Big Data. Es positivo contar con muchos datos, pero no servirán de nada a menos que se puedan monetizar. Según las estimaciones hechas por la compañía International Data Corporation (IDC) el Big Data moverá más de 187 billones de dólares en 2019 [24]. Para obtener la mayor rentabilidad posible es necesario tener claro qué datos hay que recolectar, qué procedimientos analíticos seguir y cómo transformar los datos en conocimiento y el conocimiento a su vez en valor.

2.1.2. *Big Data vs. Business Intelligence*

Talend es una de las herramientas ETL más importantes en la actualidad. En una sección de su página web oficial se declara que los procesos de extracción, transformación y carga (ETL) son componentes críticos para la alimentación de un data warehouse, un sistema Business Intelligence o una plataforma Big Data [25]. De esto último se desprende que los sistemas ETL están ligados en cierta manera tanto a Big Data como a Business Intelligence, por lo que es conveniente enfrentar estos dos conceptos para poder disipar dudas como la de si existen herramientas ETL para Big Data y otras para Business Intelligence o como la de si las herramientas de este tipo que incorporan BI son una evolución mejorada respecto a las que proceden de Big Data. Por tanto, hay que saber si Big Data sustituye a Business Intelligence o si se complementan.

Ambas tecnologías, Big Data y Business Intelligence, se utilizan para analizar datos y de esta manera ayudar a las empresas en el proceso de toma de decisiones, pero se distinguen en el modo en el que lo llevan a cabo y en el tipo de datos que analizan.

En términos generales, la diferencia consiste en que BI ayuda a encontrar respuestas a las preguntas que se conocen de antemano. Por el contrario, Big Data ayuda a encontrar las preguntas que no se conoce que se quieren pedir. Si se entra más en detalle, se puede ver que Big Data puede analizar datos en distintos formatos, tanto estructurados como semi-estructurados, cosa que no ocurre con Business Intelligence. Además, las soluciones que aporta Big Data trasladan las funciones de proceso a los datos, en vez de los datos a las funciones. Asimismo, en BI los datos suelen encontrarse en un servidor central, en el caso de Big Data se pueden encontrar en sistemas de ficheros distribuidos.

Hay que tener en cuenta que el fin último del Big Data es descubrir el conocimiento que esconden los datos para después utilizarlo en la mejora de cualquier tipo de proceso implicado en el negocio. Por consiguiente, esta tecnología no sustituye a la de Business Intelligence sino que la complementa, tratándose de una herramienta más que estará implicada a la hora de generar mejores operaciones de inteligencia de negocio que producirán valor que, como ya se vio en el apartado anterior, constituye una de las cinco magnitudes claves de Big Data.

Como ya se ha comentado, el Big Data y el Business Intelligence son complementarios, la razón se debe a que mientras la primera permite disponer de un análisis profundo y una visión global de los datos, la segunda ofrece elementos como cuadros de mando o reportes que aportan análisis fiables avanzados de una manera visualmente atractiva.

A continuación, mediante la ilustración número 4 se explica de una forma muy simple y visual las diferencias existentes entre Big Data y Business Intelligence:

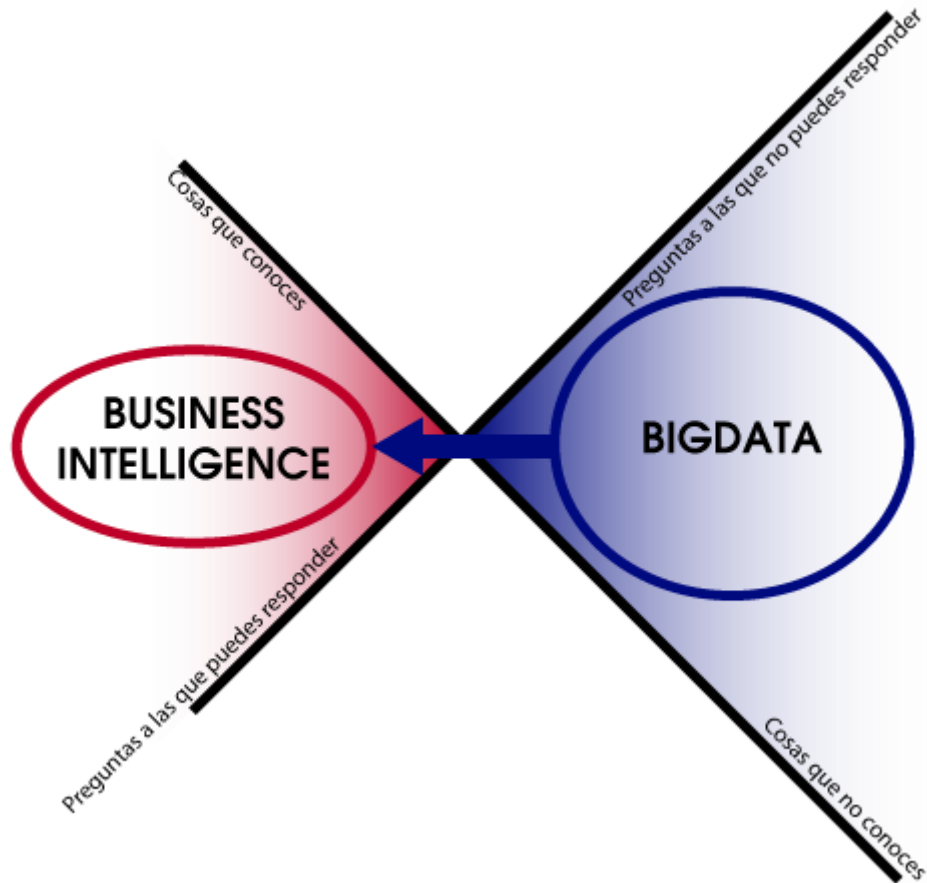


Ilustración 4 - Comparativa entre Big Data y BI

Como conclusión en base a lo anterior, se entiende que Big Data es una evolución del Business Intelligence y, por el contrario, se entiende que las herramientas ETL tradicionales formaban parte de la tecnología de Big Data y han ido incorporando funcionalidades propias de BI con el paso del tiempo. La aplicación que se encarga de la elaboración y gestión de procesos ETL fruto de este estudio no incorpora BI, aunque las tareas realizadas con ella son capaces de generar la materia prima de la que se alimenta esta tecnología encargada de transformar los datos en información. La razón fundamental es que la experiencia lleva a determinar que cuanto mayor es el grado de especialización de una herramienta, mayor es el rendimiento y calidad que se puede llegar a obtener. Es decir, en las herramientas propias de BI los productos de cuadros de mando integrales y los sistemas de soporte a la decisión, por lo general, están mejor desarrollados y ofrecen un mayor número de funcionalidades que los que pueden aportar las herramientas ETL. Basta con comparar una biblioteca de creación de informes como es JasperReports con el módulo de BI de una herramienta originalmente de ETL como Pentaho para llegar a esta conclusión.

2.2. Procesos ETL

La arquitectura de inteligencia de negocio se puede dividir en dos entornos: el primer entorno es el analítico en el cual se extraen los datos del almacén de datos y con diferentes técnicas de análisis o visualización de datos, como técnicas de minería de datos o de cuadros de comando, se presenta la información analizada. El segundo entorno se conoce con el nombre de Data Warehousing y es el encargado de extraer los datos de diferentes fuentes de información y cargarlos en el almacén de datos. Para esto, se necesita un proceso ETL que es lo que se va a tratar en este apartado.

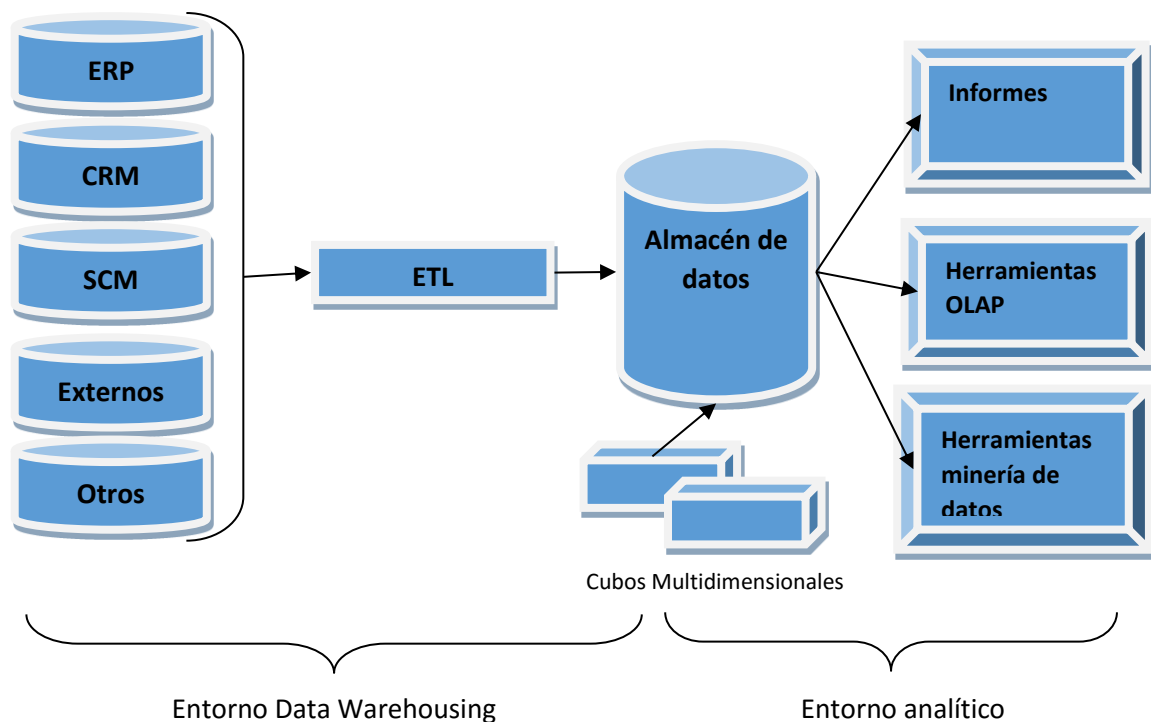


Ilustración 5 - Arquitectura de inteligencia de negocio

ETL procede de las siglas en inglés ETL = Extract – Transform – Load que traducidas al español serían Extraer - Transformar - Cargar, estas se corresponden con las tres etapas de este proceso. El objetivo principal de este proceso ETL es facilitar el movimiento de los datos y la transformación de los mismos, integrando los distintos sistemas y fuentes en la organización moderna.

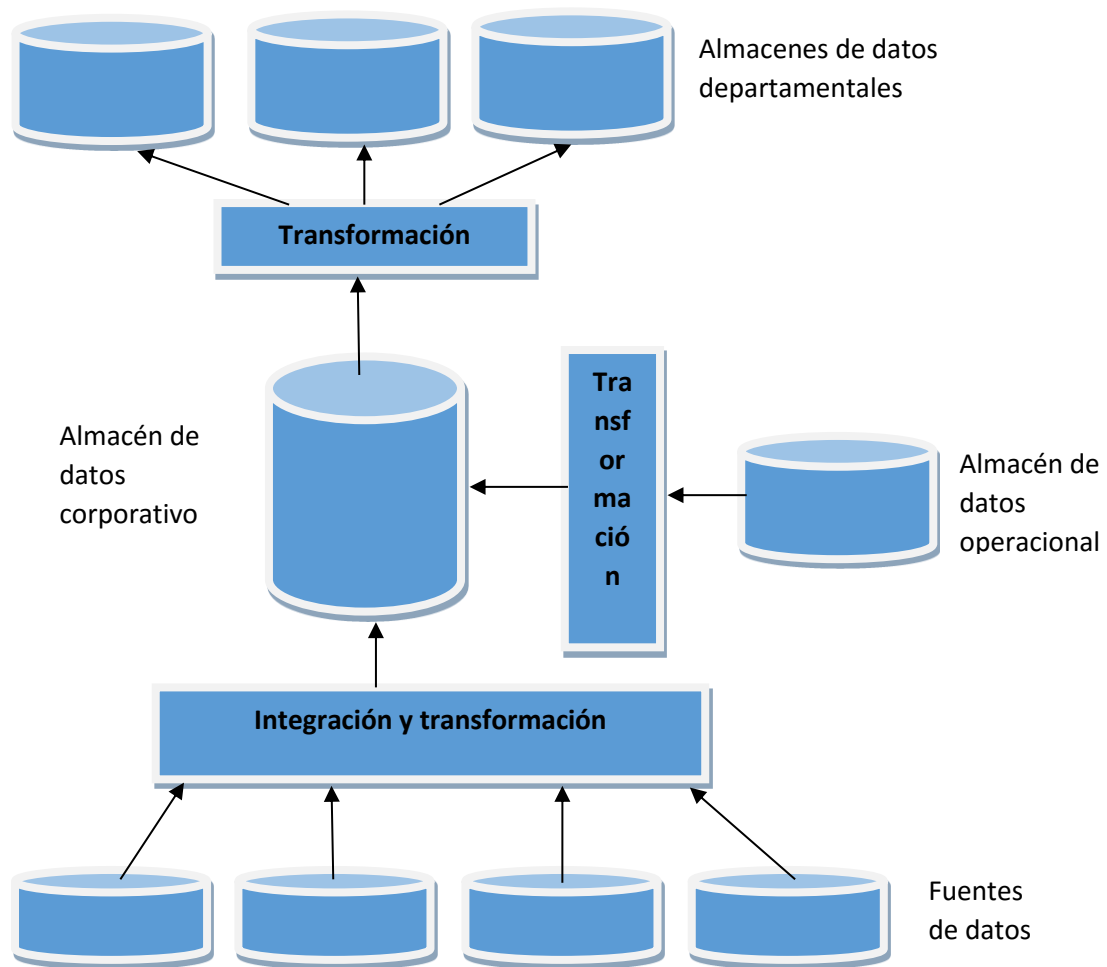


Ilustración 6 - Elementos del proceso ETL

En la ilustración número 6 acerca del proceso ETL se puede apreciar la arquitectura de inteligencia de negocio de la parte del entorno de Data Warehousing de una manera más detallada. En ella se pueden ver las distintas fuentes de datos como son los ERPs, CRMs y SCMs, que son los sistemas operacionales y también los datos externos de la organización. De igual forma, se puede observar que hay tres tipos de almacenes de datos diferentes: el almacén de datos operacional, el almacén de datos corporativo o data warehouse y el almacén de datos departamental o data mart. Seguidamente se van a presentar estos dos últimos almacenes de los que tanto se está oyendo hablar dentro del mundo de Big Data:

- **Data warehouse:** se trata de una base de datos corporativa que se ocupa de integrar y refinar la información de una o más fuentes diferentes, para posteriormente tratarla permitiendo su análisis desde un sinfín de perspectivas con un tiempo de respuesta muy satisfactorio. Agrupa los datos de toda la empresa con el fin de

facilitar su análisis, de manera que sean útiles para acceder y analizar información sobre la propia empresa.

Para el trabajo de un almacén de datos corporativo es crucial la integración de los datos que proceden de bases de datos distribuidas por las distintas unidades de la organización y que a menudo dispondrán de distintas estructuras. De lo que se trata es de ofrecer una descripción global y un análisis comprensivo de toda la empresa en el almacén de datos. Un almacén de datos de estas características se elabora a partir de los datos de las distintas fuentes de datos (fuentes heterogéneas) de una compañía, lo que implica un esfuerzo para poner en común este dato procedente de las distintas fuentes. Otra idea crucial para el trabajo de un almacén de datos corporativo es la separación de los datos utilizados en operaciones diarias de los datos utilizados en el data warehouse para los propósitos de divulgación, que facilita la toma de decisiones, para el análisis y para operaciones de control. Los objetivos de ambos tipos de datos son muy distintos, motivo por el cual no deben convivir en la misma base de datos.

Entre las ventajas que ofrece este almacén de datos se tienen servir de soporte en la toma de decisiones de la organización a cualquier nivel jerárquico y suponer un complemento esencial de la minería de datos. Por el contrario, tiene desventajas tales como disponer de una capacidad limitada e implicar un alto coste tanto su implantación como su mantenimiento. Además, se corre el riesgo de que queden obsoletos de un día para otro.

- **Data mart:** se trata de una base de datos departamental que está especializada en el almacenamiento de los datos de un área de negocio en particular. En resumidas cuentas, se trata de una versión condensada y más centrada de un almacén de datos corporativo que refleja las normas y especificaciones del proceso de cada unidad de negocio dentro de una organización, cada almacén de datos departamental a una región de negocio o región específica. Este subconjunto de datos puede extenderse a lo largo de muchas o todas las áreas funcionales de una empresa. Es común para múltiples data marts que se utilicen con el fin de servir las necesidades de cada unidad de negocio individual, distintos data marts se pueden usar para obtener información específica para los diferentes departamentos de la empresa, tales como ventas, contabilidad y un largo etcétera.

Se pueden encontrar dos tipos de data mart: OLTP y OLAP. Los sistemas OLTP, Online Transactional Processing, son bases de datos destinadas al procesamiento de transacciones que pueden conllevar operaciones de inserción, modificación y borrado de datos. El acceso a los datos se encuentra optimizado para operaciones frecuentes de lectura y escritura. Por su parte, los sistemas OLAP, Online Analytical Processing, son bases de datos destinadas al procesamiento analítico. La acción más habitual es la consulta, siendo menos habituales las inserciones, modificaciones y borrados.

Entre las ventajas que ofrece este almacén de datos se tienen centrarse en el área de trabajo en el que se desea trabajar, acelerar las consultas reduciendo el volumen de datos a recorrer, estructurar los datos para su propicio acceso por una herramienta, segmentar los datos en distintas plataformas hardware y suponer un coste muy inferior al de un almacén de datos corporativo. Por el contrario, tiene desventajas tales como la posibilidad de acabar con un maremágnum de data marts

dispares y no relacionados suponiendo una tarea tediosa su mantenimiento y la de no permitir el manejo de grandes volúmenes de información ya que no pueden ver a la organización como un todo.

Una vez entendido el entorno de Data Warehousing, se puede responder a la pregunta sobre qué hace el proceso ETL. Cada base de datos o almacén de datos tendrá asociado su propio proceso ETL que se ocupará de extraer, transformar y cargar los datos. Es decir, el almacén de datos operacional tendrá que extraer los datos de las diferentes fuentes de datos internos de la organización de los sistemas operacionales y también de los sistemas externos. Además, tendrá que transformar los datos para poder cargarlos en el almacén de datos operacional con la estructura y requerimientos que necesita o que tiene este almacén de datos operacional. Tan pronto como se tiene preparado este almacén de datos operacional se debe cargar el almacén de datos corporativo o data warehouse. La forma de hacerlo es identificando en el almacén previo cuáles son los datos que se necesitan, cuáles son las estructuras más idóneas y, finalmente, cargarlos en el almacén de datos corporativo. Éste es único, sólo hay un almacén de datos corporativo. En cambio, para cargar los diferentes almacenes de datos departamentales o data mart lo que se debe hacer es volver a extraer, identificar, obtener los datos del data warehouse, transformarlos según la estructura del data mart al que tiene que ir destinado y, por último, cargar o distribuir los datos en el almacén de datos departamental concreto.

El siguiente paso es estudiar cómo funciona cada una de las etapas del proceso ETL:

- **Proceso de extracción:**

La primera fase se corresponde con la obtención de datos. En esta etapa se deben identificar y determinar de entre todas las fuentes de datos que existen en la organización qué datos son los necesarios. En el momento en que están identificados estos datos hay que diferenciar entre la carga inicial y las actualizaciones. En un almacén de datos existen dos tipos de información: la información histórica y la información actual. El proceso ETL será diferente en función de si se trata de una imagen inicial o de una actualización.

Otra característica es la clasificación de los datos que se pueden diferenciar en datos estructurados que provienen de estos sistemas operacionales, los datos semi-estructurados como los xml y los datos no estructurados que pueden provenir de los textos, vídeos o imágenes. Por tanto, los datos se extraen de fuentes de datos heterogéneas y la integración de todos los sistemas dispares en toda la empresa supone el verdadero reto para conseguir que el almacén de datos alcance un estado en el que se pueda utilizar.

Es necesario contar con un mapa lógico de datos antes de que los datos físicos puedan ser transformados. El mapa de datos lógicos describe la relación entre los puntos del extremo inicial y los puntos del extremo final del sistema ETL presentados generalmente en una tabla:

Objetivo			Origen		Transformación
Nombre tabla	Nombre columna	Tipo dato	Nombre tabla	Nombre columna	Tipo dato

Tabla 1 - Tabla de mapa lógico de datos

El contenido del documento de mapeo de datos lógicos ha resultado ser el elemento crítico necesario para planificar de manera eficiente los procesos ETL. Donde la tabla

anterior debe representar el curso de acción que participa en el proceso de transformación.

Esta etapa de extracción debe procurar afectar en la menor medida posible al rendimiento de los sistemas de origen. Cuando se extrae un volumen de datos considerablemente grande, es posible encontrarse en la situación en la que los sistemas de origen no puedan prestar sus servicios con normalidad, viéndose mermado su rendimiento o incluso llegando al colapso. Por este motivo, es una práctica habitual cuando se trabaja con sistemas de gran tamaño programar las operaciones de extracción en fechas y momentos del día en los que el impacto sea el mínimo posible.

- **Proceso de transformación:**

La siguiente fase es la transformación, depuración e integración. La primera parte consiste en transformar estos datos que se han obtenido de los sistemas operacionales o de los datos externos en la estructura que tiene que ir en el almacén de datos, como por ejemplo:

- Modificar la estructura de los campos mediante el fusión o la división de los mismos, como por ejemplo dividir la columna “Nombre: Chamorro Rodríguez, Juan” en dos columnas: “Nombre: Juan” y “Apellidos: Chamorro Rodríguez”.
- Modificar el formato o tipo de dato, como por ejemplo el formato de fecha.
- Modificar las unidades, como por ejemplo los cambios de moneda.
- Agregar información de temporalidad, es decir, definir el período de validez de los datos.
- Modificar el alcance de la agregación, como por ejemplo calcular el salario anual de un empleado a partir de lo que cobra mensualmente.

En el caso de la depuración, lo que hay que realizar es detectar y corregir los valores que sean inconsistentes. Se debe tener presente que los datos que se cargan en los almacenes de datos tienen que ser datos de calidad, por tanto, se han de identificar errores, teniendo estos datos completos y correctos. Además, si hay valores por defecto, también se deben añadir.

La última parte será la integración, en ella se debe realizar la carga de estos datos en el almacén de datos, bien si es inicial o una actualización. El principal problema consiste en detectar que los datos representen el mismo concepto y para transformar estos datos hay que homogeneizar la representación y eliminar la información que esté duplicada.

- **Proceso de carga:**

La tercera fase supone el último paso del proceso ETL y consiste en cargar estos datos, es decir, transportar los datos de las diferentes plataformas y cargarlos en la base de datos correspondiente.

A continuación, se van a explicar las formas básicas existentes mediante las cuales se puede realizar el proceso de carga:

- **Acumulación simple:** se trata de llevar a cabo un resumen de todas las transacciones que se encuentren dentro del intervalo de tiempo en el que se quiera trabajar y llevar el resultado obtenido hacia el almacén de datos en forma de una sola transacción, guardando un valor calculado que se corresponderá, normalmente, con un sumatorio o un promedio de la magnitud tenida en cuenta.

- **Rolling:** proceso indicado en aquellas ocasiones en las que se apueste por mantener varios niveles de granularidad. Con este fin se guarda información resumida a distintos niveles que pertenecen a diferentes agrupaciones de la unidad de tiempo o con distintos niveles jerárquicos en alguna o varias de las dimensiones de la magnitud guardada, como ejemplo se tienen los totales diarios, totales semanales, mensuales, etc.

Para poder controlar un proceso ETL existen soluciones o herramientas que facilitan todo este trabajo de extraer, transformar, depurar y, finalmente, cargar. Esas herramientas ETL facilitan el trabajo automatizando la extracción de los datos. Al mismo tiempo, permiten manejar excepciones, así como planificar y realizar una gestión de la planificación de los procesos que se deben realizar para hacer la carga en el almacén de datos.

Es cierto que las herramientas ETL se emplean en la fabricación y en el mantenimiento de los almacenes de datos, pero además pueden utilizarse para desempeñar otros menesteres tales como los siguientes:

- Transportar los datos de una aplicación a otra distinta. El motivo de esta migración puede ser por cambio de versión o de aplicación.
- Trabajar sobre bases de datos pudiendo consolidar, migrar y sincronizar bases de datos operativas.
- Asegurar datos que se encuentran en gran número dentro de sistemas paralelos con el fin de conservar históricos o para poder realizar procesos de borrado en los sistemas originales.
- Hacer que coincidan en el tiempo diferentes sistemas operacionales.
- Enviar información a clientes y proveedores y, además, capturar, procesar e integrar la información recogida.
- Hacer las operaciones necesarias para poder realizar trabajos masivos, como pueden ser el envío de información por correo a las personas que forman parte de una lista establecida o el envío de un boletín de noticias a los afiliados de una empresa.

2.2.1. Herramienta ETL

En los siguientes párrafos se va a estudiar la herramienta ETL introducida en el apartado anterior. El estudio será genérico, es decir, no se pondrá el foco en ninguna herramienta del mercado en particular.

En el año 2012, la empresa multinacional InetSoft, dedicada al diseño de aplicaciones para la inteligencia de negocio, organizó una conferencia con el título "*Enterprise Data Integration Methods: ETL.*" que fue transmitida por Internet y que estuvo a cargo de Abhishek Gupta, product manager de esta empresa [37]. En esta conferencia se expuso de una manera clara y concisa el mundo de las herramientas ETL explicando los comienzos de las mismas, la evolución que han tenido, las principales ventajas y desventajas que tienen y para qué tipos de trabajo son ideales. Por estas razones, se ha tomado como base la transcripción de esta conferencia para explicar las herramientas ETL aportando un enfoque personal.

Las herramientas ETL han existido desde hace más de dos décadas. De hecho, compañías como IBM o Informatica han estado optimizando sus motores ETL desde entonces destinando cientos de millones en el desarrollo de software [38]. Se puede situar el inicio de estas herramientas en la época en la que empezaron a migrarse datos de aquellas computadoras grandes, potentes y costosas, usadas principalmente por una gran compañía para el procesamiento de una gran cantidad de datos, cuando las personas movían los datos de una aplicación a otra. Por consiguiente, esta vida tan longeva y el interés que han tenido grandes compañías tecnológicas en su desarrollo, han dado lugar a que sea una de las tecnologías de integración de datos más maduras.

Se trata de una tecnología de movimiento de datos empleada para mover los datos recibidos del almacén de datos de una aplicación a otra localización sin necesidad de interactuar con las interfaces de programación de una aplicación. Por ende, omite toda la lógica de la aplicación y se centra en la capa de datos. Y, a la sazón, se tiene un lugar de destino en el que se intentan insertar esos datos.

De esta forma se pueden leer bases de datos tan dispares como MySQL o MongoDB, archivos como XML o CSV, servicios web y transportar todos estos elementos juntos. La manera en que las herramientas simplifican esta acción consiste en conectar sus bibliotecas y almacenes de metadatos integrados por debajo de ellos. Esto provoca que la trazabilidad y el mantenimiento sean más sencillos que si se tratase de un entorno de código hecho a mano.

Estas herramientas ETL se pensaron para que desempeñaran trabajos por lotes. A este respecto, era frecuente emplear esta tecnología para elaborar procesos destinados a realizar copias de seguridad de bases de datos de las empresas. Estos procesos, normalmente, eran programados para que se ejecutasen en los momentos del día en los que no había actividad laboral. Es decir, no se requería un tiempo de respuesta tan bajo ni tener que trabajar en tiempo real, características que hoy en día sí son requeridas. Para adaptarse a estos nuevos tiempos, han ido evolucionando en recopilaciones de aplicaciones informáticas o plataformas. Las cuales disponen de una cantidad importante de componentes de integración distintos. De esta manera, se puede observar la incorporación de algunos de estos medios en tiempo real o acceso bajo demanda. Esta adaptación no ha afectado al entorno de almacenamientos de datos, donde las herramientas ETL continúan satisfaciendo las necesidades en un porcentaje muy alto.

La propia herramienta se utiliza para especificar las fuentes de datos y las reglas para la extracción y el procesamiento de dichos datos y, seguidamente, ejecutar el proceso. De esta

forma ya no será necesario tener que escribir los procedimientos y el código, tarea que podía consumir mucho tiempo y resultar muy tediosa. Por el contrario, utiliza una programación por componentes para producir componentes funcionales con interfaces bien definidas empleadas para la comunicación entre éstos [87]. Estos componentes se presentan en una interfaz gráfica en la cual se determinan las reglas y, generalmente utilizando una interfaz de arrastrar y soltar, se visualizan los flujos de datos en un proceso.

Su *modus operandi* consiste en tomar esas reglas y ejecutarlas a través de un motor o generando código en un fichero ejecutable, que posteriormente será ejecutado dentro de un entorno de producción. Como ya se ha comentado anteriormente, las herramientas ETL se pensaron para ser ejecutadas como sistemas por lotes. Por tanto, o bien existe alguna clase de evento que desencadena la acción o bien existe una programación horaria que especifica las fechas y los momentos del día en los que se lanzará el proceso. De esta forma, puede haber dependencias en la programación de tal modo que si un proceso se ejecuta de manera satisfactoria, otro pueda ser activado para ejecutarse.

En cuanto al perfil del trabajador que hace uso de estas herramientas, normalmente se trata de una persona que tiene conocimientos de bases de datos y lenguaje SQL y que entiende los datos que tiene que tratar. Es comprensible que posea estos conocimientos, ya que es frecuente tener que elaborar procesos ETL que tengan que extraer los datos de entrada de base de datos y/o cargar los datos de salida en la misma o distinta base de datos.

Esta clase de herramientas son muy útiles a la hora de mover datos a granel, conseguir grandes cantidades de datos y transferirlos por lotes. Además, funcionan bien en aquellas ocasiones en las que se tienen reglas complejas y transformaciones, siendo ideales para tratar grandes volúmenes de datos que proceden de distintos tipos de fuentes.

2.2.2. Estructuras de datos en los sistemas ETL

Potencialmente, cualquier fuente de datos estructurada o cualquier fuente de datos no estructurada en la que exista o pueda existir un algoritmo para poder estructurarla puede integrarse dentro de un sistema ETL.

A continuación, se detallan las principales estructuras de datos que soportan la mayoría de herramientas ETL que el mercado pone a disposición:

- **XML:** este sistema que permite definir lenguajes según las necesidades ofrece la posibilidad de incorporar nuevas etiquetas después de haber realizado el diseño del documento. Asimismo, facilita el análisis y el procesamiento de los documentos XML realizados por otros y, no requiere realizar cambios para cada versión del metalenguaje. Una de sus mayores ventajas es que puede poner en contacto una aplicación con otras de distintas plataformas sin tener en consideración el origen de la información que comparten. Las herramientas ETL no sólo son capaces de leer y escribir ficheros XML, sino que también permiten indicar el mapeo de campos de un componente utilizando este metalenguaje.

- **JSON:** se trata de un formato de texto de intercambio de datos ligero. Es ideal tanto para los humanos como para las máquinas, puesto que para los primeros resulta fácil de leer y escribir y para los segundos es fácil para analizar y generar. Mantiene una clase de estructuras consistente en objetos que disponen de una colección de pares llaves-valor y una clase de estructura que implementa arrays de valores. Asimismo, es independiente del lenguaje, a pesar de seguir las convenciones reconocidas por los programadores de lenguajes como C, Java o Javascript entre otros. Estas características que aporta JSON hacen que sea un lenguaje idóneo para el intercambio de datos y para las APIs web en particular.

El auge del desarrollo de aplicaciones web y esa idoneidad para trabajar con las APIs web han fomentado que JSON haya ganado popularidad a lo largo de los últimos años en detrimento de XML.

El formato de datos ligero JSON puede ser convertido al meta-lenguaje XML, y viceversa. Hoy en día existen aplicaciones cuya función es llevar a cabo esta conversión, por lo que es suficiente con que la herramienta ETL utilizada soporte estructuras de datos XML para que se pueda trabajar con estructuras de datos JSON.

- **CSV:** se trata de un tipo de documento en formato abierto en el que se disponen los datos siguiendo una estructura de tabla. Las columnas se separan utilizando normalmente punto y coma y las filas mediante saltos de líneas. Conviene reseñar que existen herramientas ETL que permiten especificar el juego de caracteres que se precise utilizar en un fichero en particular, tanto para separar las columnas como para separar las filas.

Por otra parte, este tipo de herramientas no sólo son capaces de identificar las columnas mediante símbolos identificados como separadores, sino que también permiten identificar columnas por la posición y el número de caracteres fijo que ocupen.

Las ETL, para poder tratar la información procedente de estas estructuras de datos, necesitan identificar las columnas y las filas de las tablas que representan y, además, conocer el nombre y el tipo de dato que representa cada una de estas columnas. Esta información es solicitada al usuario de la herramienta, puesto que ésta no es capaz de deducirla.

- **Servicios Web:** permiten que diferentes aplicaciones de software desarrolladas utilizando diferentes lenguajes de programación y ejecutadas sobre cualquier plataforma puedan intercambiar datos en redes de ordenadores como Internet.

WSDL es un estándar empleado por las aplicaciones ETL para poder trabajar con los servicios web. Está basado en XML y se encarga de describir los requisitos funcionales que son necesarios para poder iniciar una comunicación con los Servicios Web. Es decir, aporta información acerca de cuáles son las interfaces facilitadas por el Servicio Web y los tipos de datos requeridos para su uso.

Para que una herramienta ETL que soporte los Servicios Web pueda trabajar con éstos precisa que se le facilite, al menos, información acerca de la dirección del WSDL, datos de autenticación en el caso de ser necesarios, nombre del método y un mapeo de los parámetros.

Por último, es importante destacar que existen herramientas ETL que ofrecen la posibilidad de exportar los procesos como Servicios Web. Por ejemplo, la

herramienta de Talend Open Studio incluye entre sus tipos de exportación algunos para convertir el proceso exportado en un Servicio Web:

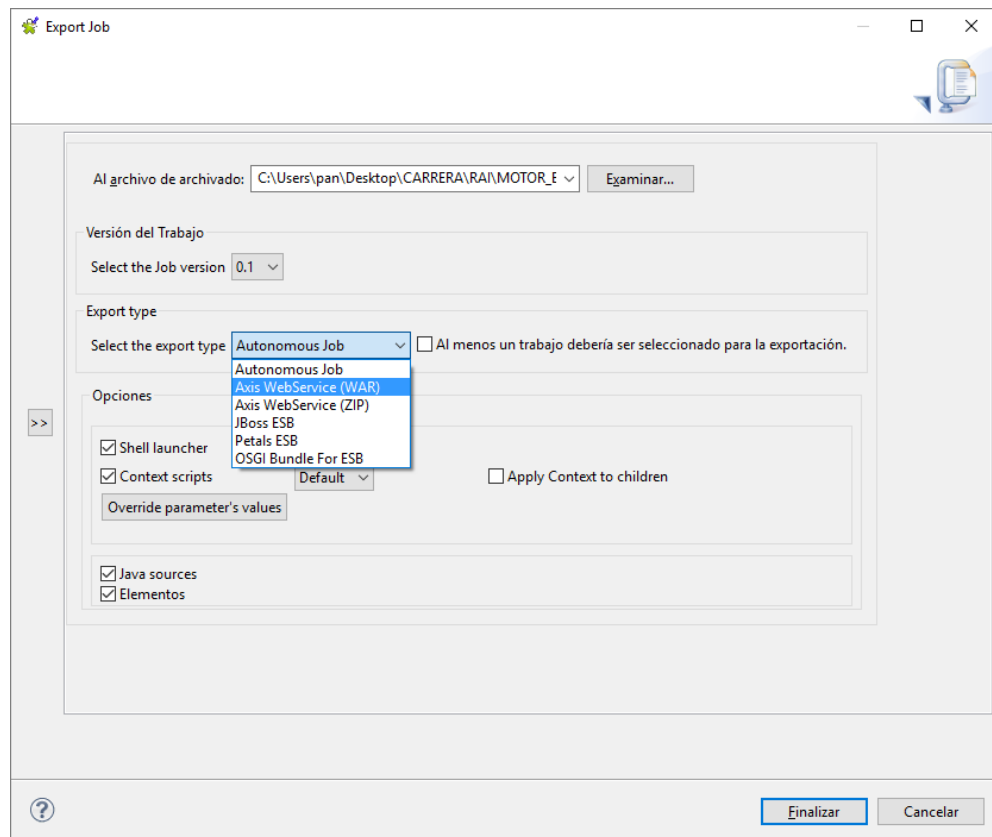


Ilustración 7 - Ejemplo exportación proceso como servicio web.

- **Bases de datos:** las principales bases de datos con las que trabajan las herramientas ETL son las denominadas relacionales y las conocidas como noSQL. Estas últimas funcionan mejor en el mundo de Big Data puesto que permiten resolver el problema de escalabilidad y rendimiento que genera el manejo de grandes volúmenes de datos. Las aplicaciones ETL más completas incorporan estas dos clases de bases de datos, puesto que la aparición de las noSQL no implica la desaparición de las bases de datos relacionales, sino que existen soluciones mejores para determinados problemas y aplicaciones.

Existen ETLs que permiten utilizar consultas del tipo Map-Reduce, pudiendo ejecutarse en todos los nodos al mismo tiempo, cada uno trabajando sobre una parte de los datos, y juntar de nuevo los resultados obtenidos para después volver a enviarlos al flujo de trabajo del proceso en cuestión.

2.2.3. Estudio de las principales herramientas ETL vigentes

Hoy en día el mercado ofrece un amplio abanico de herramientas ETL. En este apartado se analizarán las más destacadas con licencia libre y aquellas que sean una aplicación web. La razón por la que no se incluyen las herramientas de pago en este estudio se debe a que se quieren presentar solamente aquellas que cualquier persona (profesor, alumno, etc.) pueda utilizar y contrastar la información aportada. No obstante, una compañía debe tener en cuenta todas las posibilidades, teniendo que decidirse entre solución integrada o independiente, código abierto o de pago y entre la subcontratación y el desarrollo. Por tanto, los elementos que tienen que tener presente a la hora de elegir una herramienta ETL son los siguientes:

- **Tamaño de la empresa:** si se trata de una multinacional con muchas sucursales a lo largo del mundo se recomienda una solución completa y, por regla general, muy cara. Por el contrario, si se trata de una PYME se recomienda elegir soluciones de pago, ya que garantizan un cierto nivel de desahogo sin la participación de meses de desarrollo.
- **Tamaño de la estructura de TI:** una empresa con una gran organización de TI puede permitirse el lujo de optar por una solución de código abierto y personalizarlo de acuerdo a las necesidades empresariales. Sin embargo, una PYME no puede permitírselo.
- **La cultura corporativa:** en el caso de tratarse de una empresa con una cultura de código abierto muy arraigada, es muy probable que la integración de una solución de pago genere cierto rechazo.
- **Soluciones maduras:** existen soluciones bien establecidas que funcionan bien y tienen una buena regeneración, normalmente suelen ser más caras. Además, se pueden encontrar otras soluciones menos maduras que disfrutan de una cierta “moda” y que parecen ofrecer un rendimiento muy bueno. Por último, existen soluciones de código abierto que debido a su corto recorrido no ofrecen ni mucha flexibilidad ni la implementación de las soluciones citadas anteriormente. Llevará un tiempo que estas soluciones lleguen a un nivel tolerable de madurez.

2.2.3.1. Herramientas ETL con licencia libre

Este tipo de aplicaciones no poseen restricciones legales significativas en relación con el derecho de uso, la redistribución y la creación de versiones modificadas o derivadas por parte de terceros.

Los principales criterios que se han tenido en cuenta para elegir las herramientas con licencia libre que se van a estudiar a continuación, en detrimento de las demás, han sido: la usabilidad, eficiencia de los procesos ETL que pueden producir, la cantidad de tipos de fuentes de entrada y de salida que pueden manejar y el número de operaciones de transformación sobre los datos que ponen a disposición del usuario.

Talend Open Studio

La herramienta ETL desarrollada por Talend ofrece un entorno gráfico basado en Eclipse. Mediante la herramienta de diseño se elaboran los trabajos, denominados 'Jobs', gracias al paquete de componentes que ofrece. Estos Jobs se agrupan en proyectos junto con sus metadatos y contextos. Está desarrollada en el lenguaje de programación Java y permite utilizar y generar Java. Esto significa que dispone de un entorno de desarrollo multiplataforma y que puede hacer uso de este lenguaje en los trabajos de transformación de datos. Asimismo, se pueden integrar nuevos componentes mediante este lenguaje. Los Jobs realizados se transforman en código Java, que compila y facilita en forma de archivo .jar y un script .sh o .bat para poder ejecutarlos tanto en Linux como en Windows. De esta manera, la ejecución de sus procesos no depende de la herramienta. Aunque es posible ejecutarlos desde la herramienta de diseño.

Talend dispone de un número elevado de componentes que se pueden conectar mediante distintas clases de conexiones. Teniendo aquellas que se dedican al paso de información y aquellas denominadas disparadoras que permiten realizar la secuencia de ejecución y su control.

Dispone de un completo metadata que contiene las conexiones a base de datos y las tablas, vistas y consultas de ésta. También es posible definir estructuras de ficheros (delimitados, xml, etc.) que posteriormente pueden ser utilizadas de nuevo en cualquier componente. Se guardan de manera centralizada en el repositorio.

Además de los metadatos, incluye un paquete de variables conocidos como contextos que se especifican en el proyecto y que se pueden usar posteriormente en los Jobs para personalizar su forma de trabajar.

Todos los componentes de un proyecto se guardan a nivel de sistema de ficheros. Cuando se modifica un objeto, el repositorio se actualiza expandiendo la modificación a todo el proyecto.

A continuación, se van a exponer los principales aspectos positivos y negativos que presenta esta herramienta ETL:

- Al tratarse de un generador de código y seleccionar Java como lenguaje ya se ha comentado que se pueden crear nuevos componentes y utilizarlo para trabajos de transformación. El inconveniente se encuentra en que hay que dominar este lenguaje para poder explotar al máximo esta herramienta y, en ocasiones, se aprecia cierta lentitud provocada por el uso de este lenguaje.

- No resulta una tarea sencilla poder adquirir soltura y llegar a dominarla, pero la ayuda contextual que ofrece en la propia aplicación y la ayuda online de todos los componentes facilitan el proceso de aprendizaje.
- Facilita un gran número de componentes que permiten conectarse a gran variedad de sistemas y orígenes de datos. Además, en cierta medida permite diseñar componentes, gracias a que se pueden incluir librerías.
- Ofrece un análisis y actualización de dependencias cuando se producen cambios en los elementos, se reparten por todos los trabajos de un proyecto.
- Aporta un sistema de logs pudiendo mandar los registros generados a ficheros, consola o base de datos. Además, se pueden especificar tanto a nivel de proyecto como por cada trabajo en particular.
- Permite hacer debug pudiendo seguir la traza de la ejecución como si se estuviera trabajando en Eclipse.
- Permite hacer paralelismo.
- Permite volver a versiones anteriores si ocurriese algún problema.
- Para el control de flujo en los procesos se dispone por un lado de flujo de datos y por otro de disparadores para controlar la ejecución de procesos.
- Permite la gestión de errores.

Pentaho data integration (Kettle)

Se trata de un motor de transformación que ofrece una herramienta intuitiva y conceptualmente sencilla y potente. Tiene a su disposición componentes suficientes como para poder realizar la mayoría de procesos ETL.

A continuación, se van a exponer los principales aspectos positivos y negativos que presenta esta herramienta ETL:

- Ofrece un diseño de interfaz que no llama mucho la atención y no facilita una interfaz unificada en todos los componentes pudiendo ser confusa la interfaz de los componentes.
- Aunque facilita un paquete de componentes suficiente como para poder realizar la mayoría de procesos ETL, es cierto que es más limitado que el que ofrece la herramienta ETL de Talend.
- La ayuda que ofrece es mejorable, teniendo que recurrir en muchos casos a probar los componentes para poder conocer cómo funcionan.
- Ofrece un sistema de logs pudiendo mandar los registros generados a base de datos. Los logs se administran a nivel de transformaciones y trabajos.
- Permite hacer debug.
- Permite hacer paralelismo.
- Se encuentra limitado para agregar nuevas funcionalidades o modificar las que ya se tienen.
- Para el control de flujo en los procesos se pasa la información entre componentes con los saltos, de una forma única, y el flujo obtenido varía según el tipo de control. Ofrece la encapsulación de transformaciones a través de los mapping, permitiendo definir transformaciones para procesos repetitivos.

- Permite gestionar los errores pudiendo interactuar con ellos y solucionarlos antes de que termine el proceso, siempre que se esté en un paso que lo permita.

2.2.3.2. Herramientas ETL que son aplicaciones web

Poder realizar procesos ETL sobre aplicaciones web permitiría no tener que depender del sistema operativo de los equipos clientes. Además, se podría acceder a la herramienta con sólo contar con un navegador web y acceso de red al servidor donde estuviese instalada ésta, sin necesidad de tener que instalarla en cada equipo donde se fuese a utilizar. Al mismo tiempo, cualquier actualización o resolución de fallos en la aplicación no implicaría la intervención del usuario, hecho que no ocurriría si se tratase de una aplicación de escritorio, en cuyo caso cada cliente tendría que instalar la actualización de la herramienta en su equipo. Sin embargo, las herramientas ETL nacieron como aplicaciones de escritorio y esa tradición continúa a día de hoy. Por este motivo, se van a presentar las pocas herramientas ETL web relevantes que ofrece el mercado y así constatar la existencia de un nicho de mercado.



Apache Oozie

Se trata de una aplicación web implementada con el lenguaje de programación Java para operaciones de Hadoop que permite a los administradores de clústeres realizar transformaciones de datos complejas a partir de las múltiples tareas que lo componen.

Esta herramienta se utiliza para programar trabajos de Apache Hadoop. Combina varios trabajos de manera secuencial en una unidad lógica de trabajo. Se integra con la pila de Hadoop y es compatible con los trabajos de Hadoop para Apache MapReduce, Apache Pig, Apache Hive y Apache Sqoop. Además, es capaz de programar trabajos específicos para un sistema, como programas Java.

Desde la propia página web de la herramienta se destaca que se trata de un sistema que está preparado para hacerse más grande sin perder calidad en los servicios ofrecidos y el tiempo de funcionamiento sin fallos es muy alto. Asimismo, el hecho de haber sido desarrollada dentro de Apache Software Foundation hace que disponga de una buena comunidad que la soporta.

Es cierto que se trata de una aplicación web, pero está claramente ligada a Apache Hadoop. Por este motivo, se trata de una herramienta ETL muy encorsetada que no incluye, por ejemplo, componentes específicos para poder trabajar con las bases de datos relacionales principales (Oracle, MySQL, MariaDB, etc.).

Informatica Cloud

Informatica Cloud fue lanzada en 2006 por Informatica, compañía dedicada al desarrollo de software y cuyos productos están centrados en la integración de datos.

Es compatible con una gran variedad de fuentes y destinos de datos implicadas en las operaciones ETL y aprovecha su “Cloud Secure Agent” para conectar de forma segura entornos locales (hardware y software disponible en la empresa del cliente) a entornos en la nube. Entrando más en detalle, “Cloud Secure Agent” es un programa ligero que ejecuta tareas y permite la comunicación segura a través del servidor de seguridad entre la organización del cliente y el asistente de datos de Informatica Cloud.

No es necesario tener que instalar una herramienta cliente para el desarrollo, ya que se puede diseñar un proceso ETL en su totalidad mediante su aplicación web. Una vez que se ha ejecutado el proceso ETL, “Cloud Secure Agent” recupera los detalles del ETL de la nube y se ejecuta la tarea. Informatica Cloud no guarda ningún tipo de datos transaccionales en sí, sino que se dedica a servir de puente realizando la ejecución con el apoyo de “Cloud Secure Agente” y moviendo los datos.

Se trata de una herramienta de pago que incluye un gran número de funcionalidades para poder realizar todo tipo de operaciones ETL. Estaría recomendada para empresas pudientes que no quisieran embarcarse en el desarrollo de una herramienta propia que conllevara meses de desarrollo.

2.3. Resumen general de las herramientas actuales

Las herramientas ETL que se han estudiado no incluyen un módulo específico desde el que poder gestionar el trabajo. Es decir, un usuario de este tipo de aplicaciones agradecería poder anotar las tareas que tiene que realizar y conocer el estado y la prioridad de las mismas. Estas tareas deberían ser definidas por el propio usuario, pudiendo modificarlas a posteriori. La aplicación realizada durante el desarrollo de este trabajo tiene en cuenta esta necesidad.

Asimismo, se han encontrado herramientas ETL que agrupan los trabajos en áreas de trabajo a modo de directorios, pero no permiten decidir qué usuarios pueden acceder a cada una de las mismas. Sería beneficioso poder contar con una aplicación centralizada en la que sólo pudiesen acceder usuarios registrados y, más concretamente, sólo a las áreas de trabajo en las que estuviesen autorizados a trabajar. Esta necesidad ha sido cubierta en la aplicación de este proyecto.

Por otra parte, es difícil encontrar una herramienta ETL que sea una aplicación web y más aún si se restringe la búsqueda a aquellas que estén entre las más valoradas y utilizadas. En el presente documento se han comentado aquellas que se han podido encontrar hasta la fecha (29/08/2016) en las primeras páginas que devuelven los principales buscadores (Google, Bing, Yahoo). Se han encontrado sólo dos aplicaciones de este tipo: la primera con licencia libre y claramente enfocada en Hadoop, la segunda más completa, pero se trata de una herramienta de pago. Dado el número reducido de este tipo de aplicaciones y las características de las mismas, se ha diseñado e implementado una herramienta ETL como una aplicación web que ofrece aquellas características de las que adolecen las principales herramientas ETL que existen en la actualidad, es decir, la independencia del sistema operativo, acceso inmediato y desde cualquier lugar, soporte de múltiples usuarios concurrentes, actualizaciones sin requerir la intervención del usuario y mayor seguridad al evitar los problemas que puedan tener los ordenadores de los usuarios, por ejemplo, posibles virus.

En resumen, la mayoría de herramientas ETL actuales son completas y cubren las necesidades que se esperan de ellas, salvo aquellas que se han descrito y resuelto con la aplicación elaborada y detallada en los próximos apartados. Es importante señalar que no se pretende competir con el número de componentes que éstas ofrecen, el objetivo primordial es cubrir las necesidades mencionadas.

3. Gestión de proyecto software

La idea es llevar a cabo una simulación de la gestión del proyecto software realizado. La gestión emulará un proyecto real como si se realizase dentro del marco de una compañía. El fin de este apartado es poner en práctica los conocimientos adquiridos durante la carrera dentro del ámbito de la gestión de proyectos y ponerlos en práctica en este proyecto como si se tratara de un problema acontecido en el mundo real.

3.1. Alcance del proyecto

En el capítulo que precede al presente se han estudiado las distintas tecnologías que podían estar relacionadas con el proyecto que se plantea en este trabajo. Tras haber llevado a cabo el estudio anterior descrito, se empezará explicando la tecnología empleada para el desarrollo del proyecto para de esta forma establecer el contexto.

Como ya se ha visto anteriormente, el proyecto se centra en el desarrollo de una herramienta como una aplicación web que permita elaborar y gestionar procesos ETL. Más concretamente en el trabajo con ficheros de tipo CSV permitiendo realizar tareas de mapeo de campos y filtrado de registros sobre el flujo de datos. Estos procesos ETL elaborados por los usuarios podrán ser ejecutados o programados para su ejecución y subidos como distintas versiones en un repositorio. Además, para una mejor organización del trabajo los trabajos formarán parte de áreas de trabajo y el acceso a las mismas dependerá de si un usuario está vinculado a esta. Por último, permite que un usuario de la aplicación pueda organizarse su trabajo apuntando sus tareas y estableciendo el estado en el que se encuentren.

Teniendo en cuenta todo esto, se mostrará cómo el hecho de elaborar un proceso ETL desde una aplicación web en lugar de desde una de escritorio supondrá un descenso tanto del consumo de la CPU como de la memoria del equipo del usuario, además de la independencia del sistema operativo, acceso inmediato y desde cualquier lugar, soporte de múltiples usuarios concurrentes, actualizaciones sin requerir la intervención del usuario y mayor seguridad al evitar los problemas que puedan tener los ordenadores de los usuarios, por ejemplo, posibles virus.

Al mismo tiempo se mostrará cómo el hecho de compartimentar los procesos en áreas de trabajo y restringir su acceso mejorará la seguridad y la eficiencia del trabajo, esto último gracias también a la posibilidad que brinda la aplicación elaborada de que el usuario pueda organizar sus actividades de trabajo.

3.2. Plan de trabajo

3.2.1. Estudio de las principales herramientas ETL vigentes

Con el fin de aplicar un orden al proyecto se ha procedido a identificar cada una de las tareas llevadas a cabo en la implementación de la aplicación.

Seguidamente se procede a desglosar las tareas:

3.2.1.1. Estudio inicial del problema y estado del arte

Antes de comenzar con el desarrollo de la aplicación es obligado realizar un estudio previo acerca del problema que se pretende resolver. Con este fin, se realizará un análisis previo del contexto actual, es decir, un estudio acerca de la tecnología que ofrece el mercado que esté próxima al proyecto aquí tratado en cuanto a lo que ofrece.

- **Análisis del problema planteado:**

En este punto se identifican los objetivos del proyecto en cuestión. Hay que tener presente la viabilidad de estos objetivos y ser conscientes de si se pueden cumplir.

No existen apenas herramientas ETL que sean aplicaciones web y, por tanto, carecen de las ventajas que este tipo de aplicaciones ofrecen. Además, no incluyen repositorios de los trabajos en la propia herramienta y no permiten organizar las tareas a realizar por el usuario. Por consiguiente, habrá que elaborar un estudio del problema a partir de esto.

- **Estado del arte:**

La aplicación a desarrollar es fruto del estudio de las tecnologías existentes en el mercado que están dentro de su mismo ámbito, este estudio tiene lugar en este apartado y permitirá ofrecer al cliente una herramienta que ofrezca una mejora respecto de las existentes. Big Data, procesos ETL, herramientas ETL supondrán conceptos importantes a la hora de iniciar un análisis que abra el paso al ámbito de esta tecnología.

- **Conceptos asimilados y pruebas:**

Una vez estudiadas y asimiladas las tecnologías relacionadas con el mundo en el que se encuentra la aplicación a desarrollar es conveniente efectuar pruebas mediante herramientas parecidas a la que se pretende desarrollar para poder determinar qué se puede hacer a día de hoy y cuáles son los límites. De esta forma, se tendrá una guía a la hora de encarar el diseño, la arquitectura y el funcionamiento y así conseguir un mejor resultado.

Existe una gran cantidad de herramientas ETL que permiten determinar las carencias y límites de este tipo de aplicaciones y, por tanto, se tendrán en cuenta a la hora de presentar una herramienta ETL más eficiente.

3.2.1.2. Configuración del entorno de trabajo

Después de haber analizado el problema presentado y de haber realizado el estudio del arte, es necesario definir el entorno de trabajo con el que se implementará la aplicación.

La aplicación se implementará en Java, por lo que el sistema operativo no será relevante, pero se ha elegido Windows 10 64bits puesto que es con el que cuenta el desarrollador. Asimismo, se han tenido que estudiar diferentes frameworks que facilitasen la programación y ayudasen a estructurar mejor el programa. Además, buscar aquellos que permitiesen controlar el acceso al

sistema y aquellos para planificar tareas programadas. Estos frameworks utilizados empleados en la elaboración de la aplicación se detallarán en profundidad en el apartado 4.2.2 Diseño.

Se ha visto el entorno donde se va a implementar la aplicación, pero también se ha tenido que determinar en qué servidor estará desplegada ésta. Se ha seleccionado como sistema operativo Debian 8 <<Jessie>> 64bits y un servidor de aplicaciones Tomcat 8 para el despliegue de la herramienta ETL Web.

Al trabajar con Java, tanto en el servidor donde se ha implementado la aplicación como en el de despliegue de la misma se han instalado el JDK 7 y el JRE 1.7.

Por último, la información generada y consumida por la aplicación se almacena en un esquema de base de datos creada expresamente para esta aplicación. La base de datos elegida es MySQL 5.5 y está instalada incluyendo el esquema de la aplicación tanto en el equipo del desarrollador como en el servidor donde se encuentra desplegada la aplicación. De esto se desprende que una de las sub-tareas ha sido el diseño y puesta en marcha del esquema de base de datos necesario para el almacenamiento de la información generada por la aplicación desarrollada.

3.2.1.3. Planificación del trabajo

Una de las carencias encontradas en las herramientas ETL actuales es que éstas no disponen de un área específica desde la que el usuario pueda organizar su trabajo y saber el estado en el que se encuentran las mismas. Con este fin, se han tenido que estudiar distintas formas que se emplean hoy en día para lograr este objetivo. Finalmente, se ha obtenido la idea de una forma de administrar el trabajo utilizada en la elaboración de proyectos software: tablero Kanban, que fue descrito en la introducción de la presente memoria. Se ha tenido que definir el número de estados por los que puede pasar una tarea introducida por el usuario y determinar que la mejor forma de representar esos estados es por columnas, pudiendo cambiar el estado de una actividad arrastrando y soltando la misma de una a otra y pudiéndola eliminar si así fuera preciso.

3.2.1.4. Elaboración de procesos ETL

Para esta tarea se ha estudiado la forma en que se crean los componentes en las principales herramientas ETL que hacen programación por componentes. El número de componentes y funcionalidades que se pueden realizar dentro de un proceso ETL son prácticamente infinitas por lo que se ha centrado el foco en el trabajo con ficheros de tipo CSV permitiendo realizar tareas de mapeo de campos y filtrado de registros sobre el flujo de datos. Se ha estudiado que las principales herramientas ETL mantienen los datos del proceso ETL en memoria durante su construcción y que sólo cuando se guardan es cuando se almacenan en el disco duro o, en este caso, en base de datos. Por este motivo, se ha diseñado una estructura de *arrays* de componentes que guarde la relación con su componente de entrada, la relación con su componente de salida, los campos mapeados y, para el componente correspondiente, los filtros sobre los campos (componente de filtro) y las relaciones entre los campos de dos ficheros CSV de entrada (mapeo). Esta estructura mantenida en memoria desde la parte del cliente es emulada en el esquema de base de datos de la aplicación, por lo que la recuperación para mostrar los datos en la vista y el almacenamiento en base de datos serán tareas eficientes e intuitivas.

El usuario deberá aportar la ruta de los ficheros de tipo CSV que formarán parte de la entrada y el carácter separador de sus columnas en el componente específico para este tipo de ficheros. Del mismo modo, deberá indicar la ruta del fichero de tipo CSV de salida en el componente correspondiente.

En resumen, los componentes ofertados son:

- Fichero de entrada de tipo CSV.
- Fichero de salida de tipo CSV.
- Componente para el mapeo de campos de a los sumo dos ficheros de entrada.
- Componente para el filtrado de registros del flujo de datos.

Para poder representar gráficamente los componentes y sus conexiones se ha utilizado la librería de JavaScript llamada Jsplumb.

Las herramientas ETL estudiadas que ofrecen un repositorio para mantener versiones de los trabajos son sistemas externos a la propia herramienta, como Apache Subversion. Tras este estudio se ha presentado un sistema de versiones de repositorio mantenido desde el propio sistema de la aplicación y mantenido desde el esquema de base de datos de la aplicación, de tal manera que se ha creado una relación lógica entre un trabajo y la versión correspondiente.

3.2.1.5. Programación de trabajos

Una vez que se ha realizado un trabajo de tipo ETL es necesario ponerlo en funcionamiento para que desempeñe la tarea para la que fue creada. Se ofrece una vista desde la que programar la ejecución de la versión de un trabajo con ayuda de la librería de Java llamada Quartz Scheduler.

El usuario deberá indicar el trabajo, la versión del mismo y el horario de programación y la aplicación le ofrecerá información acerca de la fecha, el estado y la descripción de la última ejecución llevada a cabo.

3.2.1.6. Estructura de los trabajos

En esta tarea se ha tenido que determinar la forma en la que estructurar los trabajos (ETL) llevados a cabo por el usuario sabiendo que no todos los usuarios tienen porque tener el mismo cometido. Con este fin se ha elaborado una estructura lógica en la que los trabajos forman parte de un área de trabajo que emula a un directorio físico, pudiendo haber tantos áreas de trabajo como cree el administrador. Además, sólo se muestran al usuario aquellas áreas de trabajo a las que éste esté vinculado por lo que se logra compartimentar el trabajo y, por ende, mayor eficiencia y seguridad. Dentro del área de trabajo un usuario puede visualizar/eliminar trabajos y acceder/eliminar versiones en el repositorio del trabajo, así como acceder a su copia del trabajo.

3.2.1.7. Administración

Se ha tenido que definir el módulo donde un administrador pueda visualizar, dar de alta/baja o modificar la información de un empleado dentro de la aplicación y determinar el rol que tendrá en el sistema y si se encuentra activo o no. Así mismo, se ha tenido que definir el módulo desde

el que un administrador pueda visualizar/añadir/eliminar áreas de trabajo y vincular/desvincular usuarios a las mismas.

3.2.1.8. Control de acceso

En este punto se ha tenido que determinar que los roles de la aplicación son dos: administrador y usuario. El administrador sólo puede acceder a las vistas de administración que permiten la gestión de usuarios y áreas de trabajo. Por el contrario, los usuarios sólo pueden acceder a la vistas para la planificación del trabajo, a la vista para la elaboración de procesos ETL dentro de un área de trabajo que tenga vinculada y a la vista desde la que poder planificar la ejecución de los trabajos ETL realizados desde la propia aplicación.

Tras un estudio exhaustivo se ha concluido controlar el acceso mediante un sistema de *login* con *nick* y clave únicos por usuario. Tras el acceso se determinará el rol y se mostrará el módulo de administración (administrador) o el módulo de trabajo (usuario). En el módulo de trabajo se ofrecen sólo las áreas de trabajo a las que pertenezca un usuario, restringiendo el acceso si intentase acceder a una a la que no estuviese vinculada.

El acceso a los controladores del servidor se realiza a través de URLs y parte de ellas están restringidas para que sólo un determinado rol pueda acceder a ellas. Además, los métodos que trabajan directamente con la base de datos están anotados para que devuelvan error de permisos en el caso de que un rol no autorizado intente acceder a ellos.

3.2.1.9. Evaluación del trabajo elaborado

Tras finalizar el desarrollo de la aplicación para la elaboración y gestión de procesos ETL, es necesario refutar su buen funcionamiento y garantizar que se ajusta a lo que se esperaba de ella al comienzo del proyecto.

Con este fin, se llevará a cabo una batería de pruebas unitarias sobre la aplicación.

- **Evaluación de la administración:**

Se comprobará que el alta/baja/modificación de usuarios se refleja correctamente en el esquema de base de datos y que los datos son mostrados correctamente al usuario según la información almacenada.

Del mismo modo, se comprobará que el alta/baja de áreas de trabajo se refleja correctamente en el esquema de base de datos, así como la vinculación y desvinculación de usuarios existentes a las áreas de trabajo y que esta información es mostrada de forma correcta al usuario de la aplicación según la información almacenada.

- **Evaluación de los trabajos realizados:**

Se comprobará que cada uno de los componentes creados realiza las operaciones para las que fueron destinadas y que funciona correctamente el sistema de control de procesos ETL que informa de posibles errores en su construcción en el momento de guardarlo. Del mismo modo, se comprobará la integridad de los datos, demostrando que al recuperar un trabajo guardado, ya sea la copia del usuario o una versión de repositorio, el proceso contiene la misma información.

- **Evaluación de la ejecución de los trabajos programados:**

Se comprobará que los trabajos programados se ejecutan según la programación establecida y que se refleja el estado correcto de la última ejecución, así como la fecha

de la última ejecución y la descripción adecuada que refleje con fidelidad lo que haya ocurrido durante la ejecución del trabajo.

3.2.1.10. Memoria TFG

Por último, se realizará una memoria con el fin de detallar el trabajo llevado a cabo hasta el presente, señalando todas las etapas seguidas durante el tiempo que ha abarcado la implementación y la búsqueda de información.

3.2.2. Estimación y planificación de tareas

A partir de las tareas definidas en el punto anterior se ha realizado una sucesión de tablas para preciar el tiempo expresado en horas consumido por las distintas fases del proyecto.

Se comienza mostrando un calendario compactado para poder reflejar la estimación de tareas y se concluye con un diagrama de Gantt en base a estas tareas concebidas.

3.2.2.1. Calendario

Con el fin de conocer de manera más precisa las tareas llevadas a cabo y el tiempo empleado para cada una de ellas, en la tabla que se muestra a continuación se desgrena el tiempo empleado entre todos los hitos a realizar y que se han detallado en el punto anterior.

TAREA	FECHA INICIO	FECHA FIN	DURACIÓN HORAS	DURACIÓN DÍAS
Estudio inicial del problema y estado del arte	15/05/16	02/07/16	90	48
Configuración del entorno de trabajo	28/06/16	15/07/16	51	17
Planificación del trabajo	21/07/16	02/08/16	36	12
Elaboración de procesos ETL	03/08/16	18/08/16	98	15
Programación de trabajos	19/08/16	28/08/16	30	9
Estructura de los trabajos	29/08/16	03/09/16	18	5
Administración	04/09/16	07/09/16	12	3
Control de acceso	05/09/16	08/09/16	8	3
Evaluación del trabajo realizado	01/09/16	18/09/16	60	17
Memoria TFG	12/06/16	20/09/16	94	100
TOTAL	15/05/16	23/09/16	497	122

Tabla 2 - Calendario de tareas

El proyecto se inició el día 15 de Mayo del 2016 y concluyó el día 23 de Septiembre de ese mismo año, con un total de casi 5 meses de duración. En la tabla anterior se detalla el número de horas y de días que ha supuesto el total del proyecto, así como el tiempo dedicado a cada una de las tareas realizadas. Es conveniente aclarar que han existido festivos que también se han computado puesto que durante ese periodo se ha continuado trabajando en el proyecto.

Hay que señalar que se han tenido en cuenta las horas reales tanto de desarrollo como de búsqueda de información, pero estas actividades no se han desempeñado de forma continua durante las fechas indicadas. Esto se traduce en que ha habido días en los que no se ha realizado

ninguna actividad, pero han computado como días de trabajo. Sin embargo, estos días en los que no se ha rendido no han sumado horas en el progreso del proyecto.

3.2.2.2. Diagrama de Gantt

Mediante un diagrama de Gantt se van a planificar y programar las tareas definidas anteriormente a lo largo de un período determinado de tiempo, pudiendo visualizar las dependencias existentes entre las tareas y los intervalos de tiempo entre las mismas.

Este tipo de herramientas resultan útiles en la gestión de proyectos puesto que sirven para representar las distintas fases, tareas y actividades programadas como parte de un proyecto o para enseñar una línea de tiempo en las distintas tareas llevando a cabo la técnica más eficaz.

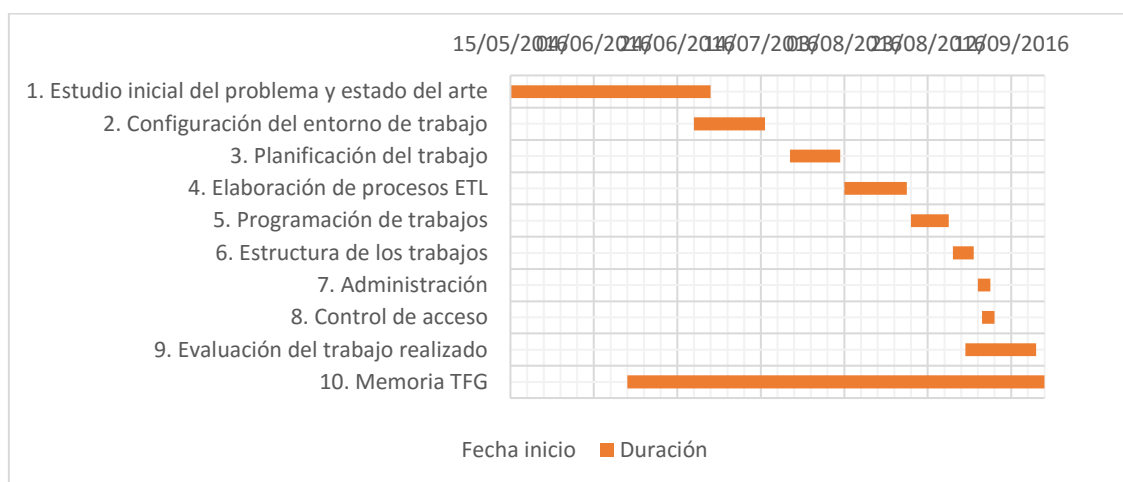


Ilustración 8 - Diagrama de Gantt

El diagrama de Gantt presentado en la ilustración anterior refleja las tareas llevadas a cabo durante todo el desarrollo del proyecto y que tiene como fecha de partida el día 15 de Mayo de 2016.

Es relevante destacar que la primera tarea con la que se arrancó el proyecto fue la dedicada al estudio inicial del problema y al estado del arte, mucho antes incluso de comenzar a redactar la memoria. Es entendible, ya que antes de iniciar la memoria hubo que determinar cuál iba a ser el tema de estudio, analizar la situación actual y encontrar debilidades para finalmente aportar soluciones. En este caso, el estudio se ha centrado en el papel que juegan los procesos ETL y las herramientas ETL empleadas dentro del mundo de Big Data en un entorno empresarial y las mejoras que se han podido aportar tras el estudio de este tipo de herramientas existentes en la actualidad.

En cuanto a las tareas centradas en el desarrollo de la aplicación se puede apreciar que la aquella dedicada a la elaboración de procesos ETL es la que ha consumido más tiempo. Esto se debe no sólo al tiempo que se ha tenido que emplear en su implementación, sino sobre todo en el estudio de las estructuras de datos utilizadas para almacenar los diferentes componentes de los procesos ETL y sus relaciones y en la mejor manera de presentárselos al usuario de la aplicación.

Se puede apreciar que las tareas destinadas al desarrollo de la aplicación se han realizado en cadena y que prácticamente no se han solapado, es decir, se ha querido terminar una tarea antes de empezar la siguiente en lugar de juntarse con muchas tareas a medio hacer. Esta forma de trabajar ha permitido mayor orden y eficiencia en la labor desempeñada.

En cuanto a la tarea destinada a la memoria del TFG se puede apreciar que ha sido la que más tiempo ha requerido abarcando prácticamente toda la vida del proyecto. Esta circunstancia se debe a que la inclusión de una tarea supone introducir información acerca de la misma en la memoria, aunque no será hasta que concluya esta tarea cuando se puedan redactar todos los detalles de la misma.

Por último, se quiere señalar que el diagrama de Gantt mostrado refleja las fechas obtenidas una vez finalizado el proyecto ajustándose a la realidad, ya que no se efectuó una planificación al comienzo al no querer reflejar fechas de finalización sin conocer todavía a ciencia cierta el planteamiento del problema y las soluciones a realizar.

3.3. Gestión de recursos

En el presente apartado se especificará el presupuesto del proyecto de la herramienta ETL Web. Para poder llevar a cabo esta labor se han tenido presentes tanto los costes materiales como los personales. Por el contrario, no se han tenido en consideración aquellos materiales que se consumen con el uso como pueden ser cuadernos en papel utilizados en las reuniones con el cliente. Del mismo modo, no se ha prestado atención a costes como el desplazamiento a la oficina del cliente. Finalmente, se mostrará también el cálculo de los costes derivados de las herramientas empleadas para la elaboración del proyecto, por ejemplo, los ordenadores usados.

Hay que tener en consideración que la unidad monetaria para expresar los costes será el Euro (€) y que el número de decimales empleados para expresar una cantidad económica será de dos, realizando la operación de truncado si fuera necesario.

3.3.1. Participantes del proyecto

Durante el período que ha conformado el desarrollo del proyecto han estado involucradas las siguientes personas:

- Juan Chamorro Rodríguez:
Estudiante de Grado en Ingeniería Informática en la Universidad Carlos III de Madrid que se ha ocupado del desarrollo completo del presente proyecto, partiendo con el estudio de las tecnologías y finalizando con la implementación de la aplicación ETL Web. Los roles que ha tomado han sido principalmente el de analista y desarrollador como Ingeniero Junior, también ha sido la cara visible de la empresa ficticia del proveedor.
- Harith Aljumaily:
Doctor en Ingeniería Informática por la Universidad Carlos III de Madrid que ha actuado como tutor del proyecto encaminando el progreso de éste y aportando conocimiento para que el desarrollo del sistema propuesto llegara a buen puerto.

A continuación, se mostrará una tabla para detallar la categoría, el coste por horas, horas totales consumidas durante el proyecto y, por último, el gasto total.

NOMBRE	CATEGORÍA	COSTE HORA	HORAS TOTALES	COSTE TOTAL
Juan Chamorro Rodríguez	Ingeniero Junior	8	497	3.976 €
Harith Aljumaily	Ingeniero	18	25	450 €
TOTAL				4.426 €

Tabla 3 - Personal encargado del proyecto

3.3.2. Costes de los elementos software y hardware

A continuación, se muestra una tabla que detalla los equipos empleados en la elaboración del proyecto, es decir, aquellos mecanismos hardware y software que se han requerido para la consecución del proyecto de forma eficiente y a tiempo.

DESCRIPCIÓN	TIEMPO DEDICADO (MESES)	UTILIZACIÓN EN %	PRECIO
PC Hewlett-Packard Intel(R) Core(TM) i7-4710HQ CPU @ 2,50GHz	5	100	958.99 €
Windows 10 Home	4	80	135 €
Debian 8 <<Jessie>>	1	20	Gratis
Eclipse Java EE IDE (<<Mars>>)	5	100	Gratis
Apache Tomcat 8	4	80	Gratis
Maven	5	100	Gratis
Spring Framework 4	5	100	Gratis
AngularJS	5	100	Gratis
Twitter Bootstrap	5	100	Gratis
Jsplumb	3	60	Gratis
Quartz (scheduler)	1	20	Gratis
TOTAL			1.093,99 €

Tabla 4 - Elementos empleados en el proyecto

3.3.3. Coste total del proyecto

Mediante la tabla de costes personales y la de materiales empleados se ha podido obtener el coste total del proyecto, es decir el importe que debe asumir la empresa contratante y que se muestra en la siguiente tabla:

Descripción	Precio
Coste de personal	4.426 €
Coste de materiales	1.093,99 €
TOTAL	5.519,99

Tabla 5- Coste total del proyecto

Por consiguiente, el coste final del proyecto es de *cinco mil quinientos diecinueve con noventa y nueve euros*.

Una vez que se ha calculado el coste final del proyecto es momento de realizar los cálculos pertinentes para obtener cierto beneficio. Con este fin, se presenta la siguiente tabla que refleja un beneficio del 15% sobre el total del coste del proyecto:

Descripción	Precio
Coste de personal	4.426 €
Coste de materiales	1.093,99 €
Beneficios (15%)	827,99 €
TOTAL	6.347,98 €

Tabla 6 - Coste total del proyecto con beneficios

Esta última tabla refleja la suma total del presupuesto del proyecto que asciende a *seis mil trescientos cuarenta y siete con noventa y ocho euros*, de los que 827,99 € se corresponden con el beneficio obtenido.

3.4. Gestión de riesgos

3.4.1. Riesgos

En el presente apartado se procederá a identificar los riesgos que puedan poner en peligro la ejecución del proyecto de la herramienta ETL Web para poder evitarlos, menguar el daño que puedan provocar o asumir un riesgo en concreto.

3.4.2. Identificación y análisis de riesgos

Para la definición de los riesgos se va a utilizar el siguiente formato de tabla:

ID	Nombre	Categoría
Descripción		
Probabilidad	Impacto	Riesgo
Consecuencia		
Solución		

Tabla 7 - Formato de tabla de recogida de riesgos

En ella se contemplan los siguientes campos:

- **ID:** Representación unívoca de cada riesgo. La nomenclatura que se emplea tendrá el formato RI – XX, las letras RI indican que es riesgo y las letras XX corresponden a la numeración que permitirá distinguir unívocamente cada uno de los riesgos.
- **Nombre:** Este campo indicará el nombre que sirve de resumen del riesgo.
- **Categoría:** Este campo indicará la categoría en la que se encuentra el riesgo.
 - **Personal** – Riesgos relacionados con el equipo de trabajo.
 - **Tecnológico** – Riesgos relacionados con los equipos informáticos que utilizarán los miembros del equipo durante el desarrollo del proyecto.
 - **Gestión** – Riesgos relacionados con la dirección del proyecto.
 - **Imprevistos** – Riesgos relacionados con imprevistos como fenómenos naturales, robos,...
 - **Explotación** – Riesgos que pueden aparecer una vez que el sistema ha sido desarrollado y es utilizado por los usuarios.
 - **Desarrollo** – Riesgos que pueden aparecer durante el desarrollo del proyecto.
- **Descripción:** Este campo tendrá una descripción detallada del riesgo.

- **Probabilidad:** Este campo tendrá un valor numérico que indicará la probabilidad que existe de que se dé el riesgo. Los posibles valores que puede tomar son los valores de 1 a 5, siendo 1 la probabilidad más baja y 5 la más alta.
- **Impacto:** Este campo tendrá un valor numérico que indicará el impacto que tendría en el proyecto en el caso de que se produjese. Los posibles valores que puede tomar son los valores de 1 a 5, siendo 1 la probabilidad más baja y 5 la más alta.
- **Riesgo:** Este campo tendrá un valor numérico que indicará el riesgo que supone, y se consigue multiplicando la probabilidad por el impacto, por lo que sus valores se encuentran en el rango 1 - 25, siendo 1 un riesgo de baja importancia y 25 la máxima.
- **Consecuencia:** En este campo indica que cosas le ocurrirán al proyecto en el caso de que se dé el riesgo.
- **Solución:** descripción breve de la posible solución frente al riesgo.

Una vez se ha definido el formato de tabla, se procede a explicar cada uno de los riesgos analizados.

ID	RI - 01	Nombre	Bajo rendimiento	Categoría	Personal
Descripción	Algún miembro del equipo no rinde como debería por cualquier motivo.				
Probabilidad	3	Impacto	3	Riesgo	9
Consecuencia	Retrasos en la realización de actividades. Desarrollo de un software de mala calidad.				
Solución	Motivar al miembro del equipo haciéndole ver la recompensa que conllevará la finalización del proyecto.				
ID	RI - 02	Nombre	Fallo del sistema informático	Categoría	Tecnológico
Descripción	Fallo del sistema informático de algún miembro o de todo el equipo.				
Probabilidad	2	Impacto	5	Riesgo	10
Consecuencia	Retrasos en la realización de actividades, lo que podría afectar a la fecha de entrega del producto.				
Solución	Disponer de equipos de repuesto.				
ID	RI - 03	Nombre	Problema con proveedor ISP	Categoría	Tecnológico
Descripción	Problema del proveedor ISP contratado, que impide que los sistemas se puedan conectar a internet.				
Probabilidad	3	Impacto	4	Riesgo	12
Consecuencia	Retrasos en la realización de actividades, lo que podría afectar a la fecha de entrega del producto.				
Solución	Hablar con el proveedor o negociar con otro.				

ID	RI - 04	Nombre	Pérdida de archivos	Categoría	Tecnológico
Descripción	Pérdida de archivos, ya sea por robo, accidente, o cualquier causa.				
Probabilidad	2	Impacto	5	Riesgo	10
Consecuencia	Retrasos en la realización de actividades, lo que podría afectar a la fecha de entrega del producto. Además, si se supone de una pérdida de datos importantes podría afectar también al presupuesto del proyecto.				
Solución	Realizar una copia de resguardo periódica de los archivos				
ID	RI - 05	Nombre	Virus	Categoría	Tecnológico
Descripción	Entrada de un virus en los sistemas informáticos de la empresa.				
Probabilidad	3	Impacto	4	Riesgo	12
Consecuencia	Retrasos en la realización de actividades, al tener que limpiar todos los equipos afectados , siempre y cuando los virus no provoquen fallos en el hardware, afectando económicamente al proyecto.				
Solución	Mantener actualizados los sistemas de seguridad.				
ID	RI - 06	Nombre	Modificación de requisitos	Categoría	Gestión
Descripción	Modificación de los requisitos, tras encontrar algún error, o alguna característica que no satisface las necesidades del cliente.				
Probabilidad	3	Impacto	5	Riesgo	15
Consecuencia	Posibles retrasos en la realización de actividades. Si la modificación se produce en etapas tempranas del proyecto tiene menos impacto que si se produce en etapas más tardías. Además, del coste económico que supondría dependiendo de la etapa en la que se realiza el cambio.				
Solución	Establecer una buena comunicación con el cliente para una buena definición de los requisitos.				
ID	RI - 07	Nombre	Mala estimación del esfuerzo	Categoría	Gestión
Descripción	Estimación del esfuerzo errónea, ya sea una estimación en la que sobran horas de trabajo, o faltan horas de trabajo.				
Probabilidad	3	Impacto	4	Riesgo	12
Consecuencia	Posibles retrasos en la realización de actividades, llegando a tener que modificar los costes del proyecto en el caso de que la estimación del esfuerzo fuera escasa al necesitar más horas de trabajo.				

Solución		Recalcular la estimación del esfuerzo lo antes posible para minimizar los daños.			
ID	RI - 08	Nombre	Mal cálculo de costes	Categoría	Gestión
Descripción		Cálculo de costes erróneo al haberse cumplido uno o varios riesgos que influyan en el coste del proyecto, provocando la modificación del presupuesto del proyecto.			
Probabilidad		3	Impacto	5	Riesgo 15
Consecuencia		Pérdidas o ganancias de dinero no esperadas.			
Solución		Recalcular los costes lo antes posible para minimizar los daños.			
ID	RI - 09	Nombre	Retrasos en las entregas	Categoría	Gestión
Descripción		Entrega de documentos en fechas que no son las establecidas al no haber completado las actividades en el tiempo estimado.			
Probabilidad		3	Impacto	3	Riesgo 9
Consecuencia		Pérdida de confianza del cliente, mala imagen de la empresa.			
Solución		Planificar mejor los recursos y tareas y realizar las futuras entregas a tiempo.			
ID	RI - 10	Nombre	Mala comunicación con el cliente	Categoría	Gestión
Descripción		Falta de comunicación con el cliente.			
Probabilidad		3	Impacto	5	Riesgo 15
Consecuencia		Se puede crear un producto que no es el que desea el cliente.			
Solución		Reavivar la relación con el cliente lo antes posible.			
ID	RI - 11	Nombre	Fallo eléctrico	Categoría	Imprevistos
Descripción		Fallo eléctrico en la oficina, por motivos ajenos a la empresa.			
Probabilidad		1	Impacto	3	Riesgo 3
Consecuencia		Retraso del proyecto hasta que el fallo sea reparado.			
Solución		Intentar buscar otras instalaciones desde las que poder seguir trabajando.			
ID	RI - 12	Nombre	Catástrofes naturales	Categoría	Imprevistos
Descripción		Catástrofes naturales como inundaciones, terremotos, incendios, que pueden llegar a destruir la oficina de trabajo y los avances del proyecto.			
Probabilidad		1	Impacto	5	Riesgo 5

Consecuencia	Dependiendo de la gravedad puede suponer un gasto económico, o la desaparición de la empresa.				
Solución	Intentar buscar otras instalaciones desde las que poder seguir trabajando.				
ID	RI - 13	Nombre	Robo	Categoría	Imprevistos
Descripción	Robo de material de la oficina o de datos.				
Probabilidad	1	Impacto	5	Riesgo	5
Consecuencia	Retrasos en la realización de actividades y aumento del coste del proyecto al tener que contratar seguros y sistema de alarmas.				
Solución	Guardar el material en un lugar seguro.				
ID	RI - 14	Nombre	Vulnerabilidades	Categoría	Explotación
Descripción	Aparición de vulnerabilidades no detectadas en el sistema.				
Probabilidad	2	Impacto	3	Riesgo	6
Consecuencia	Potencialmente graves: pérdida de datos y funcionalidad.				
Solución	Hacer un estudio de las posibles vulnerabilidades y adelantarse a ellas.				
ID	RI - 15	Nombre	Bugs	Categoría	Explotación
Descripción	Aparición de bugs o errores no detectados en el sistema.				
Probabilidad	2	Impacto	3	Riesgo	6
Consecuencia	Coste económico necesario para solventarlos. Si se han pasado las pruebas de sistema, poco probables o consecuencias menores. La imagen de la compañía y del producto puede verse afectada.				
Solución	Realizar una batería de pruebas lo más completa posible.				
ID	RI - 16	Nombre	Espionaje industrial	Categoría	Explotación
Descripción	Algún agente externo (otra empresa) se hace con datos de la aplicación o sus usuarios.				
Probabilidad	1	Impacto	5	Riesgo	5
Consecuencia	Coste económico y de imagen del producto.				
Solución	Instalar la aplicación y su base de datos en un servidor seguro.				
ID	RI - 17	Nombre	Mala comunicación entre módulos de la aplicación	Categoría	Desarrollo
Descripción	Problemas en la comunicación entre los distintos módulos desarrollados.				

Probabilidad	2	Impacto	5	Riesgo	10
Consecuencia	Retrasos en la entrega del producto final, además del posible aumento del coste dependiendo de la etapa en la que se detecte el error.				
Solución	Realizar una batería completa de pruebas.				
ID	RI - 18	Nombre	Brecha de seguridad	Categoría	Desarrollo
Descripción	Aparición de brechas de seguridad que puedan ser utilizadas para la obtención de los datos que gestiona el sistema. Nombres de empleados y sus correos electrónicos.				
Probabilidad	2	Impacto	5	Riesgo	10
Consecuencia	Problemas legales al poder exponer datos personales sin su consentimiento.				
Solución	Asegurarse que sólo usuarios con rol de administrador pueden acceder a esta información.				

Tabla 8 - Tabla de riesgos

3.5. Plan de pruebas

En este apartado se detallarán las pruebas que van a ser efectuadas para corroborar si la funcionalidad de la aplicación realiza de forma correcta las tareas que se han expuesto anteriormente.

Es necesario aclarar que las pruebas se irán implementando según se vaya completando la implementación de la aplicación, lo que se traduce en que existirán pruebas que no podrán ser tratadas hasta concluir el desarrollo del proyecto.

3.5.1. Definición y especificación de pruebas

Con el fin de llevar un mejor control sobre las pruebas se han dividido en las siguientes clases:

- Pruebas de planificación de actividades de trabajo
- Pruebas de gestión y elaboración de trabajos
- Pruebas de programación y ejecución de trabajos
- Pruebas de administración

Para cada una de esas clases de prueba se han llevado a cabo las siguientes pruebas:

- **Pruebas unitarias:**
Se han tenido en cuenta este tipo de pruebas al trabajar con programación orientada a objetos. La intención es redactar casos de prueba para cada método en el módulo, de manera que cada caso sea independiente del resto. Se emplean para corroborar el buen funcionamiento de un módulo del código, lo que se traduce en confirmar que cada uno de los módulos trabaje de forma correcta por separado.
- **Pruebas funcionales:**

Este tipo de pruebas se sustenta en la ejecución, revisión y retroalimentación de las funcionalidades anteriormente diseñadas para el software. Es decir, son pruebas concretas y exhaustivas para probar y validar que el software realiza el cometido que debe hacer y que lo hace según lo especificado.

- **Pruebas de rendimiento:**

Estas pruebas se llevan a cabo para saber cómo de rápido la aplicación resuelve una tarea en condiciones particulares de trabajo. Asimismo, permiten validar la escalabilidad, fiabilidad y empleo de los recursos.

En este caso se va a determinar la velocidad y escalabilidad de la aplicación en lo referente a la ejecución de los trabajos ETL por medio de ficheros de entrada CSV de distintos tamaños. Se efectuarán distintas ejecuciones modificando el número de campos de los ficheros y las operaciones de mapeo y filtrado sobre estos.

Para poder explicar las pruebas efectuadas se va a emplear el siguiente formato de tabla:

ID	P X-Y	Fecha		Resultado	
Finalidad					
Pasos					

Tabla 9 - Formato de pruebas

A continuación, se indica el significado de cada una de sus columnas:

- **ID:** identificador de la prueba
 - **P:** valor fijo, se refiere a Prueba.
 - **X:** número de la prueba.
 - **Y:** sub-tarea.
- **Fecha:** fecha de realización de la prueba.
- **Resultado:** si la prueba ha salido bien o ha existido algún problema.
- **Finalidad:** explicación concisa de la prueba y de su propósito.
- **Pasos:** proceso seguido para poder realizar la prueba.

Pruebas de planificación de actividades de trabajo:

Estas pruebas están destinadas a comprobar el correcto funcionamiento del módulo destinado al tablero de planificación de actividades de trabajo del usuario de la aplicación.

ID	P 1-1	Fecha	02/08/16	Resultado	Satisfactorio
Finalidad		Acceder a módulo de planificación de actividades de trabajo.			
Pasos		1. Loguearse con rol de usuario. 2. Seleccionar la pestaña "Tablero Kanban". 3. Comprobar que aparece el tablero de actividades de trabajo.			

ID	P 1-2	Fecha	02/08/16	Resultado	Satisfactorio
Finalidad		Visualizar tablero de planificación de trabajo.			
Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de usuario. 2. Seleccionar la pestaña "Tablero Kanban". 3. Comprobar que las etiquetas de actividades de trabajo están en las columnas de estados que corresponden según la última vez que se guardó el tablero. 			
ID	P 1-3	Fecha	02/08/16	Resultado	Satisfactorio
Finalidad		Añadir etiqueta de trabajo.			
Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de usuario. 2. Seleccionar la pestaña "Tablero Kanban". 3. Hacer clic en el botón con el símbolo más junto a la cabecera "Entrada" para abrir el panel para añadir nueva actividad. 4. Introducir descripción de la actividad en el campo "Contenido" del panel y pulsar el botón "Añadir". 5. Comprobar que en la columna de estado "Entrada" se ha añadido la actividad con el contenido indicado. 			
ID	P 1-4	Fecha	02/08/16	Resultado	Satisfactorio
Finalidad		Eliminar etiqueta de trabajo.			
Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de usuario. 2. Seleccionar la pestaña "Tablero Kanban". 3. Hacer clic en el botón con el símbolo de una papelera asociado a la actividad a eliminar. 4. Confirmar que la actividad ha desaparecido de la vista. 			
ID	P 1-5	Fecha	02/08/16	Resultado	Satisfactorio
Finalidad		Cambiar estado de etiqueta de trabajo.			
Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de usuario. 2. Seleccionar la pestaña "Tablero Kanban". 3. Hacer clic sobre la cabecera de una actividad, sin soltar el ratón desplazar la etiqueta a otra columna de estado y soltar el ratón (arrastrar y soltar). 4. Comprobar que la actividad ha pasado de la columna de estado de origen a la columna de estado de destino. 			
ID	P 1-6	Fecha	02/08/16	Resultado	Satisfactorio
Finalidad		Refrescar tablero de planificación de trabajo.			

Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de usuario. 2. Seleccionar la pestaña “Tablero Kanban”. 3. Mover de columna de estado una actividad. 4. Hacer clic en el botón “Refrescar”. 5. Comprobar que el tablero queda igual a como estaba antes de hacer los cambios. 			
ID	P 1-7	Fecha	02/08/16	Resultado	Satisfactorio
Finalidad		Guardar cambios en tablero de planificación de trabajo.			
Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de usuario. 2. Seleccionar la pestaña “Tablero Kanban”. 3. Hacer clic sobre la cabecera de una actividad, sin soltar el ratón desplazar la etiqueta a otra columna de estado y soltar el ratón (arrastrar y soltar). 4. Hacer clic en el botón con el símbolo de una papelera asociado a una actividad para eliminarla. 5. Hacer clic en el botón con el símbolo más junto a la cabecera “Entrada” para abrir el panel para añadir nueva actividad. 6. Introducir descripción de la actividad en el campo “Contenido” del panel y pulsar el botón “Añadir”. 7. Hacer clic en el botón “Guardar”. 8. Hacer clic en el botón “Refrescar” y comprobar que se mantienen los cambios guardados. 			

Tabla 10 - Pruebas de planificación de actividades de trabajo

Pruebas de gestión y elaboración de trabajos:

Estas pruebas están destinadas a comprobar la correcta realización de los procesos ETL, es decir, el correcto mapeo de datos, filtrado de registros y relaciones entre los distintos componentes y demás funcionalidades soportadas. Además, comprobarán la correcta organización de los trabajos en áreas de trabajo y el acceso a esta información.

ID	P 2-1	Fecha	03/09/16	Resultado	Satisfactorio
Finalidad		Acceder al módulo de áreas de trabajo.			
Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de usuario. 2. Seleccionar la pestaña “Áreas de trabajo”. 3. Comprobar que la sección se llama “Seleccione área de trabajo”. 			
ID	P 2-2	Fecha	03/09/16	Resultado	Satisfactorio
Finalidad		Acceder a un área de trabajo.			

Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de usuario. 2. Seleccionar la pestaña "Áreas de trabajo". 3. Seleccionar área de trabajo del campo selector etiquetado como "Área seleccionada". 4. Comprobar que las áreas de trabajo que se listan están realmente vinculadas al usuario de la aplicación. 5. Hacer clic en el botón "Seleccionar". 6. Comprobar que la sección a la que se ha accedido se llama "Área de trabajo : [nombre del área de trabajo seleccionada]". 			
ID	P 2-3	Fecha	03/09/16	Resultado	Satisfactorio
Finalidad		Añadir trabajo a área de trabajo.			
Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de usuario. 2. Seleccionar la pestaña "Áreas de trabajo". 3. Seleccionar área de trabajo del campo selector etiquetado como "Área seleccionada". 4. Hacer clic en el botón "Seleccionar". 5. Hacer clic en el botón "Añadir trabajo" en la vista del área de trabajo previamente seleccionada para abrir panel para añadir un nuevo trabajo. 6. Indicar nombre y descripción de trabajo en los campos habilitados para ello. 7. Hacer clic en botón "Añadir". 8. Esperar mensaje informativo. 9. Comprobar que en el campo selector de trabajos aparece el nuevo trabajo. 10. Comprobar que se han creado los registros correspondientes del nuevo trabajo en base de datos. 			
ID	P 2-4	Fecha	03/09/16	Resultado	Satisfactorio
Finalidad		Eliminar trabajo de área de trabajo.			

Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de usuario. 2. Seleccionar la pestaña "Áreas de trabajo". 3. Seleccionar área de trabajo del campo selector etiquetado como "Área seleccionada". 4. Hacer clic en el botón "Seleccionar". 5. Seleccionar trabajo del campo selector etiquetado como "Seleccione un trabajo de la lista". 6. Hacer clic en el botón "Eliminar trabajo". 7. Hacer clic en el botón "Eliminar" del panel de confirmación. 8. Esperar mensaje informativo. 9. Comprobar que el trabajo ha desaparecido de la vista y del campo selector de trabajos. 10. Comprobar en base de datos que se han realizado las operaciones correspondientes para eliminar el trabajo de su área de trabajo. 			
ID	P 2-5	Fecha	03/09/16	Resultado	Satisfactorio
Finalidad		Buscar trabajo de área de trabajo.			
Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de usuario. 2. Seleccionar la pestaña "Áreas de trabajo". 3. Seleccionar área de trabajo del campo selector etiquetado como "Área seleccionada". 4. Hacer clic en el botón "Seleccionar". 5. Seleccionar trabajo del campo selector etiquetado como "Seleccione un trabajo de la lista". 6. Comprobar que en la vista se ha cargado un trabajo con el nombre y la descripción del trabajo seleccionado. 			
ID	P 2-6	Fecha	03/09/16	Resultado	Satisfactorio
Finalidad		Buscar versión de trabajo de repositorio.			
Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de usuario. 2. Seleccionar la pestaña "Áreas de trabajo". 3. Seleccionar área de trabajo del campo selector etiquetado como "Área seleccionada". 4. Hacer clic en el botón "Seleccionar". 5. Seleccionar trabajo del campo selector etiquetado como "Seleccione un trabajo de la lista". 6. Seleccionar versión del trabajo existente del campo selector etiquetado como "Seleccione una versión" en la vista del trabajo cargada. 7. Comprobar que la descripción de la versión se corresponde con la misma. 			
ID	P 2-7	Fecha	03/09/16	Resultado	Satisfactorio

Finalidad		Eliminar versión de trabajo de repositorio.			
Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de usuario. 2. Seleccionar la pestaña "Áreas de trabajo". 3. Seleccionar área de trabajo del campo selector etiquetado como "Área seleccionada". 4. Hacer clic en el botón "Seleccionar". 5. Seleccionar trabajo del campo selector etiquetado como "Seleccione un trabajo de la lista". 6. Seleccionar versión del trabajo existente del campo selector etiquetado como "Seleccione una versión" en la vista del trabajo cargada. 7. Hacer clic en el botón "Eliminar versión repositorio". 8. Hacer clic en el botón "Eliminar" del panel de confirmación. 9. Esperar mensaje informativo. 10. Comprobar que la versión ha desaparecido de la vista y del campo selector de versiones. 11. Comprobar en base de datos que se han realizado las operaciones correspondientes para eliminar la versión de trabajo. 			
ID	P 2-8	Fecha	18/08/16	Resultado	Satisfactorio
Finalidad		Elaborar proceso ETL completo en trabajo.			

Pasos	<ol style="list-style-type: none"> 1. Loguearse con rol de usuario. 2. Seleccionar la pestaña "Áreas de trabajo". 3. Seleccionar área de trabajo del campo selector etiquetado como "Área seleccionada". 4. Hacer clic en el botón "Seleccionar". 5. Seleccionar trabajo del campo selector etiquetado como "Seleccione un trabajo de la lista". 6. Seleccionar versión del trabajo existente del campo selector etiquetado como "Seleccione una versión" en la vista del trabajo cargada. 7. Hacer clic en el botón "Abrir trabajo repositorio". 8. Comprobar que se ha accedido al área de elaboración de procesos ETL del trabajo seleccionado. 9. Hacer clic en el botón "FicheroEntrada" del panel de componentes. 10. Hacer clic en el área de trabajo. 11. Comprobar que se ha creado un componente "FicheroEntrada". 12. Hacer doble clic sobre el componente "FicheroEntrada" para abrir su panel de gestión. 13. Seleccionar pestaña "Mapeo campos" del panel. 14. Introducir nombre del campo y seleccionar tipo de campo. 15. Hacer clic en botón "Añadir". 16. Seleccionar pestaña "Ver campos" y comprobar que se muestra el campo introducido. 17. Seleccionar pestaña "Fichero". 18. Introducir ruta completa del fichero CSV y campo separador de columna en los campos habilitados. 19. Hacer clic en botón "Aplicar". 20. Hacer clic en botón "Cerrar" para cerrar panel de "FicheroEntrada". 21. Hacer clic en el botón "FicheroEntrada" del panel de componentes. 22. Hacer clic en el área de trabajo. 23. Comprobar que se ha creado un componente "FicheroEntrada". 24. Hacer doble clic sobre el componente "FicheroEntrada" para abrir su panel de gestión. 25. Seleccionar pestaña "Mapeo campos" del panel. 26. Introducir nombre del campo y seleccionar tipo de campo. 27. Hacer clic en botón "Añadir". 28. Introducir el nombre de otro campo y seleccionar su tipo de campo. 29. Hacer clic en botón "Añadir". 30. Seleccionar pestaña "Ver campos" y comprobar que se muestran los dos campos introducido. 31. Seleccionar pestaña "Fichero". 32. Introducir ruta completa del fichero CSV y campo separador de columna en los campos habilitados.
-------	--

33. Hacer clic en botón “Aplicar”.
34. Hacer clic en botón “Cerrar” para cerrar panel de “FicheroEntrada”.
35. Hacer clic en el botón “Mapeo” del panel de componentes.
36. Hacer clic en el área de trabajo.
37. Comprobar que se ha creado un componente “Mapeo”.
38. Hacer clic en el botón “Filtro” del panel de componentes.
39. Hacer clic en el área de trabajo.
40. Comprobar que se ha creado un componente “Filtro”.
41. Hacer clic en el botón “FicheroSalida” del panel de componentes.
42. Hacer clic en el área de trabajo.
43. Comprobar que se ha creado un componente “FicheroSalida”.
44. Hacer doble clic sobre el primer componente “FicheroEntrada” para abrir su panel de gestión.
45. En la pestaña “Conector” del panel seleccionar el nombre del componente “Mapeo” del campo selector de conectores de salida.
46. Hacer clic en el botón “Aplicar”.
47. Hacer doble clic sobre el segundo componente “FicheroEntrada” para abrir su panel de gestión.
48. En la pestaña “Conector” del panel seleccionar el nombre del componente “Mapeo” del campo selector de conectores de salida.
49. Hacer clic en el botón “Aplicar”.
50. Comprobar que los dos componentes “FicheroEntrada” están conectados por sus correspondientes flechas al componente “Mapeo”.
51. Hacer doble clic sobre el componente “Mapeo” para abrir su panel de gestión.
52. Seleccionar la pestaña “Mapeo campos” del panel.
53. Comprobar que aparecen mapeados los campos de los dos componentes “FicheroEntrada” a los que está conectados.
54. Hacer clic en el botón con el símbolo de una papelera asociado al tercer registro que se corresponde con el segundo campo mapeado en el segundo componente “FicheroEntrada”.
55. Comprobar que ha desaparecido el campo borrado de la vista de mapeo de campos.
56. Seleccionar la pestaña “Relaciones”.
57. Seleccionar en el primer selector con los campos del primer componente “FicheroEntrada” el primer campo.
58. Seleccionar en el segundo selector con los campos del segundo componente “FicheroEntrada” el primer campo.
59. Seleccionar el tipo de relación “Left join” o “Inner join”.
60. Hacer clic en el botón con el símbolo de confirmación.
61. Comprobar que se ha añadido el registro con la relación de los campos de los ficheros de entrada.
62. Seleccionar la pestaña “Conector”.

	<ol style="list-style-type: none"> 63. Seleccionar el nombre del componente “Filtro” del campo selector de conectores de salida. 64. Hacer clic en el botón “Aplicar”. 65. Comprobar que existe una flecha que parte del componente “Mapeo” hasta el componente “Filtro”. 66. Hacer doble clic sobre el componente “Filtro” para abrir su panel de gestión. 67. Seleccionar la pestaña “Mapeo campos”. 68. Comprobar que los campos mapeados son los mismos que los que tiene mapeados el campo “Mapeo” al que está conectado. 69. Seleccionar la pestaña “Filtros”. 70. Seleccionar el campo correspondiente al primer campo mapeado en el primer componente “FicheroEntrada” en el campo selector de campos. 71. Seleccionar si el campo va a ser igual o distinto a un valor en el segundo campo selector. 72. Escribir el valor a comparar con el del campo indicado en el tercer campo descrito como “filtro”. 73. Hacer clic en el botón con el símbolo de confirmación. 74. Comprobar que el filtro se ha añadido a la vista. 75. Seleccionar la pestaña “Conector” 76. Seleccionar el nombre del componente “FicheroSalida” del campo selector de conectores de salida. 77. Hacer clic en el botón “Aplicar”. 78. Comprobar que existe una flecha que parte del componente “Filtro” hasta el componente “FicheroSalida”. 79. Hacer doble clic sobre el componente “FicheroSalida” para abrir su panel de gestión. 80. Seleccionar la pestaña “Ver campos”. 81. Comprobar que los campos mapeados son los mismos que los que tiene mapeados el campo “Filtro” al que está conectado. 82. Seleccionar la pestaña “Fichero”. 83. Indicar la ruta del fichero CSV de salida en el campo etiquetado como “Ruta fichero”. 84. Hacer clic en el botón “Aplicar”. 85. Hacer click en el botón “Guardar”. 86. Esperar mensaje informativo. 87. Pulsar botón “Salir trabajo” situado en la cabecera. 88. Seleccionar el mismo trabajo del campo selector etiquetado como “Seleccione un trabajo de la lista”. 89. Hacer clic en el botón “Abrir trabajo usuario”. 90. Comprobar que los componentes, sus datos y sus relaciones coinciden con las que se guardaron en la copia del usuario. 91. Comprobar en base de datos que se han realizado las operaciones correspondientes para guardar la copia de usuario del trabajo con el proceso ETL realizado.
--	--

ID	P 2-9	Fecha	03/09/16	Resultado	Satisfactorio
Finalidad		Subir versión de trabajo a repositorio.			
Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de usuario. 2. Seleccionar la pestaña "Áreas de trabajo". 3. Seleccionar área de trabajo del campo selector etiquetado como "Área seleccionada". 4. Hacer clic en el botón "Seleccionar". 5. Seleccionar trabajo del campo selector etiquetado como "Seleccione un trabajo de la lista". 6. Seleccionar versión del trabajo existente del campo selector etiquetado como "Seleccione una versión" en la vista del trabajo cargada. 7. Hacer clic en el botón "Abrir trabajo repositorio". 8. Comprobar que se ha accedido al área de elaboración de procesos ETL del trabajo seleccionado. 9. Hacer clic en el botón "Repositorio" para abrir panel de repositorio del trabajo. 10. Seleccionar número de versión del campo selector etiquetado como "Versión". 11. Escribir la descripción de la versión en el campo etiquetado como "Descripción". 12. Hacer clic en el botón "Subir". 12. Hacer clic en el botón "Confirmar" del panel de confirmación. 13. Esperar mensaje informativo. 14. Pulsar botón "Salir trabajo" situado en la cabecera. 15. Seleccionar el mismo trabajo del campo selector etiquetado como "Seleccione un trabajo de la lista". 16. Seleccionar la versión anteriormente creada del campo selector etiquetado como "Seleccione una versión" en la vista del trabajo cargada. 17. Hacer clic en el botón "Abrir trabajo repositorio". 18. Comprobar que los componentes, sus datos y sus relaciones coinciden con las que se guardaron en la versión. 19. Comprobar en base de datos que se han realizado las operaciones correspondientes para guardar la versión de trabajo con el proceso ETL realizado. 			

Tabla 11 - Pruebas de gestión y elaboración de trabajos

Pruebas de programación y ejecución de trabajos:

Estas pruebas están destinadas a comprobar el correcto funcionamiento de la programación y ejecución de los trabajos ETL. Comprobando que un trabajo realiza correctamente la labor para la que fue creada y que la ejecución se realiza en el momento indicado en la programación. Además, comprueban que la información que aporta el sistema acerca de la última ejecución de un trabajo sea correcta.

ID	P 3-1	Fecha	28/08/16	Resultado	Satisfactorio
Finalidad		Acceder a módulo de programación de trabajos.			
Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de usuario. 2. Seleccionar la pestaña "Programación". 3. Comprobar que la sección se llama "Programación de procesos". 			
ID	P 3-2	Fecha	28/08/16	Resultado	Satisfactorio
Finalidad		Programar la ejecución de una versión de un trabajo.			
Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de usuario. 2. Seleccionar la pestaña "Programación". 3. Seleccionar trabajo del campo selector "Nombre trabajo". 4. Seleccionar versión de trabajo del campo selector "Versión". 5. Hacer clic en botón con el símbolo de confirmación. 6. Esperar al mensaje de confirmación de alta. 7. Comprobar que se ha añadido un registro con los datos de la programación del trabajo en la misma vista. 8. Comprobar en base de datos que se han añadido correctamente los registros correspondientes. 			
ID	P 3-3	Fecha	28/08/16	Resultado	Satisfactorio
Finalidad		Eliminar la programación de la ejecución de una versión de un trabajo.			
Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de usuario. 2. Seleccionar la pestaña "Programación". 3. Hacer clic en el botón con el símbolo de la papelera asociado a la programación a eliminar. 4. Esperar al mensaje de confirmación de borrado. 5. Comprobar que se ha eliminado el registro con los datos de la programación eliminada de la propia vista. 6. Comprobar en base de datos que se han eliminado correctamente los registros correspondientes. 			
ID	P 3-4	Fecha	28/08/16	Resultado	Satisfactorio
Finalidad		Ejecutar programación de una versión de un trabajo.			

Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de usuario. 2. Seleccionar la pestaña "Programación". 3. Hacer clic en el botón con el símbolo <i>play</i> asociado al trabajo a ejecutar. 4. Esperar al mensaje de ejecución de trabajo. 5. Consultar los campos "Última ejecución", "Estado" y "Descripción" del trabajo ejecutado. 6. Comprobar físicamente que la información reflejado en esos campos se corresponde con la tarea realizada por el trabajo ejecutado: posibles errores de ejecución, resultado del fichero de salida generado según lo programado en el proceso ETL y hora de ejecución. 			
ID	P 3-5	Fecha	28/08/16	Resultado	Satisfactorio
Finalidad		Consultar información de última ejecución de los trabajos.			
Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de usuario. 2. Seleccionar la pestaña "Programación". 3. Consultar los campos "Última ejecución", "Estado" y "Descripción" de un trabajo ejecutado. 4. Comprobar físicamente que la información reflejado en esos campos se corresponde con la tarea realizada por el trabajo ejecutado: posibles errores de ejecución, resultado del fichero de salida generado según lo programado en el proceso ETL y hora de ejecución. 			

Tabla 12- Pruebas de programación y ejecución de trabajos

Pruebas de administración:

Este tipo de pruebas se centran en el área de administración para la gestión de las áreas de trabajo y de los usuarios de la aplicación. Se asume que el usuario ha accedido a la aplicación con rol de Administrador.

ID	P 4-1	Fecha	07/09/16	Resultado	Satisfactorio
Finalidad		Acceder al área de administración.			
Pasos		1. Loguearse con rol de administrador.			
ID	P 4-2	Fecha	07/09/16	Resultado	Satisfactorio
Finalidad		Visualizar áreas de trabajo.			
Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de administrador. 2. Seleccionar la pestaña "Áreas de trabajo". 3. Comprobar que aparece la información de las áreas de trabajo de forma paginada y por filas. 			
ID	P 4-3	Fecha	07/09/16	Resultado	Satisfactorio
Finalidad		Buscar área de trabajo.			
Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de administrador. 2. Seleccionar la pestaña "Áreas de trabajo". 3. Seleccionar nombre de área de trabajo en el campo selector descrito como "Introduzca nombre area". 4. Comprobar que en el listado de áreas de trabajo sólo aparece la seleccionada. 			
ID	P 4-4	Fecha	07/09/16	Resultado	Satisfactorio
Finalidad		Añadir área de trabajo.			
Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de administrador. 2. Seleccionar la pestaña "Áreas de trabajo". 3. Hacer clic en el botón "Añadir área" para abrir el panel destinado a añadir un nuevo área de trabajo. 4. Insertar nombre del nuevo área en el campo etiquetado como "Nombre". 5. Hacer clic en el botón "Añadir". 6. Esperar mensaje informativo. 7. Seleccionar nombre del nuevo área de trabajo añadido en el campo selector descrito como "Introduzca nombre area". 8. Comprobar que en el listado de áreas de trabajo aparece el nuevo área de trabajo. 			
ID	P 4-5	Fecha	07/09/16	Resultado	Satisfactorio

Finalidad		Eliminar área de trabajo.			
Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de administrador. 2. Seleccionar la pestaña "Áreas de trabajo". 3. Seleccionar nombre del área de trabajo a eliminar en el campo selector descrito como "Introduzca nombre area". 4. Hacer clic en el botón con el símbolo de la papelera asociado al área de trabajo seleccionada. 5. Hacer clic en el botón "Eliminar" del panel de confirmación de borrado. 6. Seleccionar nombre del área de trabajo eliminado en el campo selector descrito como "Introduzca nombre area". 7. Comprobar que no se obtiene resultado de la búsqueda. 			
ID	P 4-6	Fecha	07/09/16	Resultado	Satisfactorio
Finalidad		Visualizar usuarios de un área de trabajo.			
Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de administrador. 2. Seleccionar la pestaña "Áreas de trabajo". 3. Seleccionar nombre del área de trabajo a visualizar en el campo selector descrito como "Introduzca nombre area". 4. Hacer clic en el botón "Usuarios" asociado al área de trabajo para abrir panel de usuarios. 5. Comprobar aparecen los nombres de los usuarios vinculados con el área de trabajo seleccionado. 			
ID	P 4-7	Fecha	07/09/16	Resultado	Satisfactorio
Finalidad		Vincular usuario a área de trabajo.			
Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de administrador. 2. Seleccionar la pestaña "Áreas de trabajo". 3. Seleccionar nombre del área de trabajo a visualizar en el campo selector descrito como "Introduzca nombre area". 4. Hacer clic en el botón "Usuarios" asociado al área de trabajo para abrir panel de usuarios. 5. Seleccionar usuario por el nombre en el campo selector etiquetado como "Nombre de usuario". 6. Hacer clic en el botón "Añadir". 7. Comprobar que el usuario se ha añadido a la lista de usuarios del área de trabajo en el panel. 			
ID	P 4-8	Fecha	07/09/16	Resultado	Satisfactorio
Finalidad		Desvincular usuario de área de trabajo.			

Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de administrador. 2. Seleccionar la pestaña "Áreas de trabajo". 3. Seleccionar nombre del área de trabajo a visualizar en el campo selector descrito como "Introduzca nombre area". 4. Hacer clic en el botón "Usuarios" asociado al área de trabajo para abrir panel de usuarios. 5. Hacer clic en el botón con el símbolo de la papelera asociado al usuario a desvincular del área de trabajo. 6. Hacer clic en el botón "Eliminar" del panel de confirmación de borrado. 7. Comprobar que el usuario ha desaparecido de la lista de usuarios del área de trabajo en el panel. 			
ID	P 4-9	Fecha	07/09/16	Resultado	Satisfactorio
Finalidad		Visualizar usuarios.			
Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de administrador. 2. Seleccionar la pestaña "Usuarios". 3. Comprobar que aparece la información de los usuarios de forma paginada y por filas. 			
ID	P 4-10	Fecha	07/09/16	Resultado	Satisfactorio
Finalidad		Buscar usuario.			
Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de administrador. 2. Seleccionar la pestaña "Usuarios". 3. Seleccionar nombre del usuario en el campo selector descrito como "Introduzca nombre usuario". 4. Comprobar que en el listado de usuarios sólo aparece el seleccionado. 			
ID	P 4-11	Fecha	07/09/16	Resultado	Satisfactorio
Finalidad		Filtrar usuarios.			
Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de administrador. 2. Seleccionar la pestaña "Usuarios". 3. Hacer clic en el botón desplegable "Filtro". 4. Seleccionar una opción del filtro de "Usuarios activos" ('Activos', 'No activos'). 5. Seleccionar una opción del filtro de "Rol" ('Administrador', 'Usuario', 'Todos'). 6. Comprobar que en el listado de usuarios se muestran aquellos que cumplen las condiciones de los filtros. 			
ID	P 4-12	Fecha	07/09/16	Resultado	Satisfactorio
Finalidad		Añadir usuario.			

Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de administrador. 2. Seleccionar la pestaña "Usuarios". 3. Hacer clic en el botón "Añadir usuario" para abrir el panel destinado a añadir un nuevo usuario. 4. Insertar nick, nombre, correo y rol del nuevo usuario en los campos etiquetados como "Nick", "Nombre", "Correo" y "Roles" respectivamente. 5. Hacer clic en el botón "Añadir". 6. Esperar mensaje informativo. 7. Seleccionar nombre del nuevo usuario añadido en el campo selector descrito como "Introduzca nombre usuario". 8. Comprobar que en el listado de usuarios aparece el nuevo usuario. 			
ID	P 4-13	Fecha	07/09/16	Resultado	Satisfactorio
Finalidad		Modificar usuario.			
Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de administrador. 2. Seleccionar la pestaña "Usuarios". 3. Seleccionar nombre del usuario a modificar en el campo selector descrito como "Introduzca nombre usuario". 4. Hacer clic en el botón con el símbolo de edición asociado al usuario seleccionado para abrir el panel de edición. 5. Modificar nombre, roles y/o estado (activo o no activo) mediante los campos etiquetados como "Nombre", "Roles", "Estado" respectivamente. 6. Hacer clic en el botón "Editar". 7. Esperar mensaje informativo. 8. Seleccionar nombre del usuario modificado en el campo selector descrito como "Introduzca nombre usuario". 9. Comprobar que aparece el usuario con la información editada. 			
ID	P 4-14	Fecha	07/09/16	Resultado	Satisfactorio
Finalidad		Añadir área de trabajo a usuario.			
Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de administrador. 2. Seleccionar la pestaña "Usuarios". 3. Seleccionar nombre del usuario a visualizar en el campo selector descrito como "Introduzca nombre usuario". 4. Hacer clic en el botón "Áreas" asociado al usuario para abrir panel de áreas de trabajo. 5. Seleccionar área por el nombre en el campo selector etiquetado como "Área de trabajo". 6. Hacer clic en el botón "Añadir". 7. Comprobar que el área de trabajo se ha añadido a la lista de áreas del usuario en el panel. 			

ID	P 4-15	Fecha	07/09/16	Resultado	Satisfactorio
Finalidad		Desvincular área de trabajo de usuario.			
Pasos		<ol style="list-style-type: none"> 1. Loguearse con rol de administrador. 2. Seleccionar la pestaña "Usuarios". 3. Seleccionar nombre del usuario a visualizar en el campo selector descrito como "Introduzca nombre usuario". 4. Hacer clic en el botón "Áreas" asociado al usuario para abrir panel de áreas de trabajo. 5. Hacer clic en el botón con el símbolo de la papelera asociado al área de trabajo a desvincular del usuario. 6. Hacer clic en el botón "Eliminar" del panel de confirmación de borrado. 7. Comprobar que el área de trabajo ha desaparecido de la lista de áreas de trabajo del usuario en el panel. 			

Tabla 13 - Pruebas de administración

3.5.2. Evaluación final de las pruebas

Todas las pruebas han resultado satisfactorias tal y como se esperaba, puesto que es fundamental que el software esté completamente operativo y que no presente ningún tipo de fisuras. No sería razonable presentar una batería de pruebas en la que existiese alguna prueba que no se hubiese superado con éxito.

Se ha volcado el esfuerzo en las pruebas funcionales, como la elaboración de procesos ETL y la programación y ejecución de los mismos. La prueba de la elaboración de un proceso ETL ha sido lo más completa posible incluyendo todos los componentes disponibles, es decir, se han incluido dos ficheros de entrada de tipo CSV relacionados en un componente de mapeo por alguno de sus campos mapeados, para el posterior filtrado de registros comparando uno de los campos mapeados con un cierto valor y guardando el flujo de datos resultante en un fichero de salida de tipo CSV.

A este respecto, se han empleado ficheros de distinto tamaño para ver la eficacia y velocidad de los procesos realizados desde la aplicación y comparado los resultados con los ofrecidos por otras herramientas ETL para el mismo trabajo. Los resultados han sido positivos, puesto que se han logrado resultados competentes (siempre obteniéndose mejores resultados con las otras herramientas), aunque es cierto que no se han podido utilizar ficheros de tipo CSV lo suficientemente grandes como para extraer conclusiones definitivas.

Por otra parte, se ha prestado especial atención en comprobar que los componentes, las relaciones entre ellos y las tareas realizadas por cada uno de ellos se almacenen y se recuperen correctamente del esquema de la base de datos de la aplicación. Muchas herramientas ETL utilizan un sistema de ficheros para almacenar esta información, pero en el caso de la aplicación desarrollada se emplea una base de datos, siendo el motor que hace que funcione el sistema: desde el almacenamiento y la recuperación de un proceso ETL hasta la programación y ejecución del mismo.

Las pruebas sobre la organización de los trabajos y el acceso a los mismos han resultado muy positivas constatando su eficacia. Esto mismo ha ocurrido con las pruebas realizadas sobre el tablero de planificación de actividades de trabajo de los usuarios de la aplicación.

Como conclusión final acerca de las pruebas se puede decir que prácticamente han sido positivas atendiendo a todos los baremos, de lo que se desprende que se ha logrado un software perfectamente capaz de satisfacer las necesidades para las que fue diseñado.

4. Solución

En este capítulo se detallan todos los pasos seguidos en la solución que se ha realizado, incluida la batería de pruebas llevada a cabo tras la conclusión de la implementación del código.

4.1. Definición del problema: soluciones

El trabajo elaborado se fundamenta en el diseño e implementación de una herramienta ETL como una aplicación web que permita elaborar procesos ETL mediante programación por componentes, pudiendo gestionarlos y ejecutarlos. Más concretamente, en el trabajo con ficheros de tipo CSV permitiendo realizar tareas de mapeo de campos y filtrado de registros sobre el flujo de datos. Para ello será necesario trabajar con este tipo de ficheros y diseñar e implementar una serie de componentes que se encarguen cada uno de una de esas tareas. Además, será necesario realizar una estructura que establezca las conexiones entre los distintos componentes de un proceso ETL que mantenga la integridad del flujo de datos pudiendo almacenarlo y recuperarlo.

La independencia del sistema operativo, un acceso inmediato y desde cualquier lugar, soporte de múltiples usuarios concurrentes, actualizaciones sin requerir la intervención del usuario y mayor seguridad al evitar los problemas que puedan tener los ordenadores de los usuarios, por ejemplo, posibles virus son requisitos importantes que no son cubiertas por las actuales herramientas y a las que se ha dado solución mediante el desarrollo de una herramienta ETL como aplicación web.

Inciendo en este aspecto, la tecnología de hoy en día permite trabajar el intercambio de información a través de la red de una manera rápida, segura y eficaz, por lo que este tipo de aplicaciones resuelve más soluciones que los problemas que pueda conllevar. Además, la tendencia de los últimos años es trabajar desde la nube. Por ejemplo, hace unos años no era habitual editar un documento de texto desde la nube, hoy en día es una práctica frecuente y cada vez más extendida. Por este motivo, a un usuario que trabaje con herramientas ETL no le va a suponer ningún rechazo trabajar desde un navegador web, es más, se va a sentir más cómodo.

Por otra parte, trabajar en la elaboración y gestión de procesos ETL plantea una serie de interrogantes a las que hay que dar respuesta en forma de solución. Las principales preguntas son las siguientes:

- **¿Qué sucede cuando se quiere guardar o recuperar la versión de un proceso ETL?**
Los equipos de trabajo tienen que recurrir habitualmente a software externo a las propias herramientas ETL para crear un repositorio de procesos ETL, lo que supone otro sistema más a mantener y que puede provocar que se pierdan cambios al olvidarse de subir los cambios a este repositorio.
La solución consiste en ofrecer un sistema de versiones de los trabajos en la propia aplicación almacenado en el esquema de base de datos de la misma (pudiendo realizar copias de respaldo de este esquema para mayor seguridad). A la hora de acceder a un trabajo bastará con elegir la versión del mismo y, a la hora de guardarlo, sólo se requerirá indicar el número de versión.
- **¿Cómo se organiza las tareas a realizar un equipo de trabajo dedicado a realizar procesos ETL?**

La respuesta está en que no todos los equipos se organizan de la misma manera y muchos de ellos no saben cómo hacerlo. Existen casos en los que la organización deja mucho que desear, provocando retrasos en el trabajo e incluso problemas entre los propios miembros del equipo.

Las herramientas ETL que se han estudiado no incluyen un módulo específico desde el que poder gestionar el trabajo. Se ofrece como solución a los integrantes de estos equipos un espacio en el que poder anotar las tareas que tienen que realizar y conocer el estado y la prioridad de las mismas.

Por otro lado, los equipos dedicados a estos menesteres cada vez son más numerosos y especializados y esto no lo tienen en cuenta las herramientas ETL actuales. Por este motivo y a modo de solución se presenta una aplicación centralizada en la que sólo puedan acceder usuarios registrados y, más concretamente, sólo a las áreas de trabajo en las que estén autorizados a trabajar. Esta forma de organizar los trabajos ETL en áreas de trabajo facilitará poder contar con una buena organización.

- **¿Cómo se gestionan las ejecuciones de los procesos ETL?**

Normalmente, un proceso ETL necesita ser ejecutado en ciertos momentos del día o en fechas concretas en las que no se puede ejecutar de manera manual. Para ello, se recurre a aplicaciones (ej. Crontab en Linux) para lanzar estos trabajos. Esto se traduce en que se necesita otra herramienta más para programar los procesos y otra para comprobar el resultado de las ejecuciones.

La solución aportada consiste en poder programar y ejecutar los trabajos ETL desde la propia herramienta habilitando un área específica para este fin.

4.2. Proceso de desarrollo

El cometido de este apartado es detallar el proceso completo de desarrollo de la aplicación ETL Web elaborada, tomando como punto de partida la fase de análisis y finalizando con la batería de pruebas llevada a cabo.

Para este proceso se empleará el Modelo En Cascada que ofrece la Ingeniería del Software y que se ilustra mediante este diagrama:

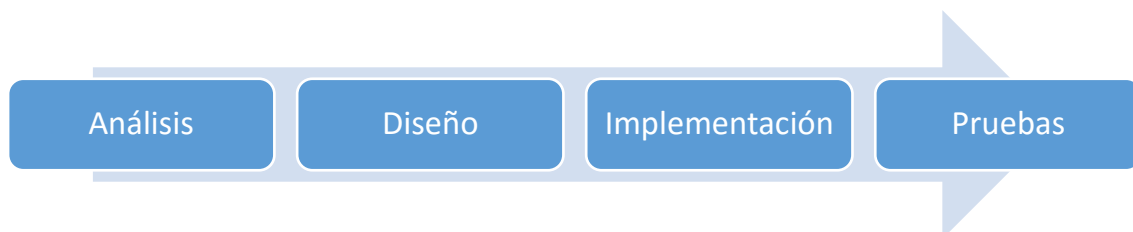


Ilustración 9 - Proceso de desarrollo por etapas

A continuación se explican cada una de las fases seguidas de manera secuencial:

- **Análisis:**
Es la fase en la que se congregan todos los requisitos que tiene que cumplir el sistema. En esta etapa es importante contar con el cliente, encargándose de documentar y analizar estos requisitos.
- **Diseño:**
Es la fase en la que se realiza un esbozo de lo requerido, documentando la interfaz y el manejo de datos del software.
- **Implementación:**
Es la fase en la que se implementa el diseño del software en el lenguaje de programación y con los frameworks detallados en el propio diseño.
- **Pruebas:**
Es la fase encargada de determinar que no se hayan generado errores en etapas precedentes, haciendo especial hincapié en la codificación.

Hay que reseñar que las fases anteriores asumen las siguientes:

- **Ingeniería del Sistema:**
Análisis de las características y el comportamiento del sistema en el que se va a integrar el software.
- **Utilización:**
Una vez superadas las pruebas, el software se pone en manos del cliente.
- **Mantenimiento:**
El software tendrá que ser modificado durante su vida útil debido a posibles errores detectados, cambios en alguno de los elementos del sistema o por nuevos requisitos que quiera incorporar el cliente.

4.2.1. Análisis

Tras haber planteado el problema a solucionar y el proceso de desarrollo que implica, el siguiente paso es iniciar la etapa de análisis.

En esta etapa se van a reunir todos los requisitos que debe cumplir el software para que éste pueda cumplir con la solución planteada. Con este fin, se ha contado con la colaboración del cliente, papel asumido por el tutor del proyecto.

Una vez especificados los requisitos recopilados, se van a detallar las secuencias de interacciones desarrolladas entre el sistema y los actores en respuesta a los eventos iniciados por un actor principal sobre el propio sistema. Es decir, se va a especificar la comunicación y el comportamiento del sistema gracias a su interacción con los usuarios. Para esto, se van a realizar casos de uso.

4.2.1.1. Definición de requisitos

Los requisitos de cada uno de los módulos se describirán en forma tabular según se puede ver en la tabla que se muestra seguidamente. Se ha decidido seguir esta estructura de ordenación para mostrar todos los requisitos de manera clara y concisa de forma que posteriormente puedan testearse.

Hay que señalar que se ha acordado con el cliente emplear frameworks específicos para el diseño de aplicaciones web que contienen plantillas de diseño con formularios, botones, cuadros, menús de navegación y otros elementos de diseño. Estas plantillas han sido aprobadas por el cliente, por lo que no se van a incluir los requisitos centrados en la presentación de las interfaces.

Identificador	Categoría		
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Riesgo	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Descripción			
Requisitos relacionados			

Tabla 14 - Formato tabla descripción de requisitos

A continuación se describen los campos:

- **Identificador:** es el código identificativo, unívoco, de cada uno de los requisitos. La nomenclatura utilizada será: **RXYZ-NNN**. En el caso de que un requisito fuera finalmente descartado, se añadirá a su identificador la etiqueta '**[ELIMINADO]**'.
- El significado de cada una de las letras es el siguiente:
- **R:** indica que es un requisito
 - **X:** indica el tipo de requisito a elegir entre los siguientes tipos:
 - **F:** requisito de funcionalidad: describe la operación que debe realizar la aplicación.
 - **R:** requisito de restricción: indica cómo debe realizarse la operación.

- **Y:** indica el módulo aplicable entre los siguientes:
 - **A:** módulo de administración
 - **G:** general/acceso
 - **T:** módulo de trabajo
 - **Z:** sub-módulos dentro de los anteriores
 - **A:** sus sub-módulos son:
 - **A:** gestión de áreas de trabajo
 - **U:** gestión de usuarios
 - **T:** sus sub-módulos son:
 - **E:** programación de trabajos ETL
 - **G:** gestión de trabajos ETL
 - **P:** planificación de trabajo
 - **NNN:** es el número de requisito dentro de la clasificación.
- **Categoría:** área/módulo al que se corresponde el requisito.
 - **Prioridad:** indica el nivel de atención al requisito y la necesidad de satisfacción del mismo. Hay tres niveles:
 - **Alto:** el requisito debe ser satisfecho antes que otros requisitos de prioridad más baja.
 - **Medio:** el requisito con este nivel debe ser satisfecho después de haber cumplido con los de nivel alto.
 - **Bajo:** requisito que, en virtud de su nivel de prioridad, es el último en satisfacerse.
 - **Riesgo:** indica la relación del requisito con su dificultad y necesidad dentro del sistema. Tiene también tres niveles, como los indicados en el punto anterior.
 - **Verificabilidad:** indica la facilidad de comprobación que se cumple el requisito. Tiene tres niveles:
 - **Alto:** el requisito puede comprobarse mediante todas las herramientas de verificación.
 - **Medio:** el requisito no puede comprobarse mediante todas las herramientas de verificación aplicables.
 - **Bajo:** requisito que no puede verificarse mediante las herramientas disponibles.
 - **Descripción:** descripción completa del requisito indicando todos sus aspectos y características.
 - **Requisitos relacionados:** indica el identificador de los requisitos con los que está relacionado si los hubiera, si no este campo quedará en blanco.

Se describirán los requisitos en modo descendente, es decir, de módulos a sub-módulos para facilitar un diseño e implementación descendente así como facilitar la comprensión de los mismos.

Dado el número total de requisitos obtenidos, las tablas de requisitos se incluyen como anexo en el apartado "[Anexo I. Requisitos](#)".

4.2.1.2. Casos de uso

En este apartado se especificarán los casos de uso. Éstos consisten en una descripción de las actividades que deberán realizarse para llevar a cabo el uso de la aplicación software por parte de los usuarios que pueden acceder a la misma.

A continuación, se presenta un esquema de los casos de uso a rasgos generales del sistema de la herramienta ETL:

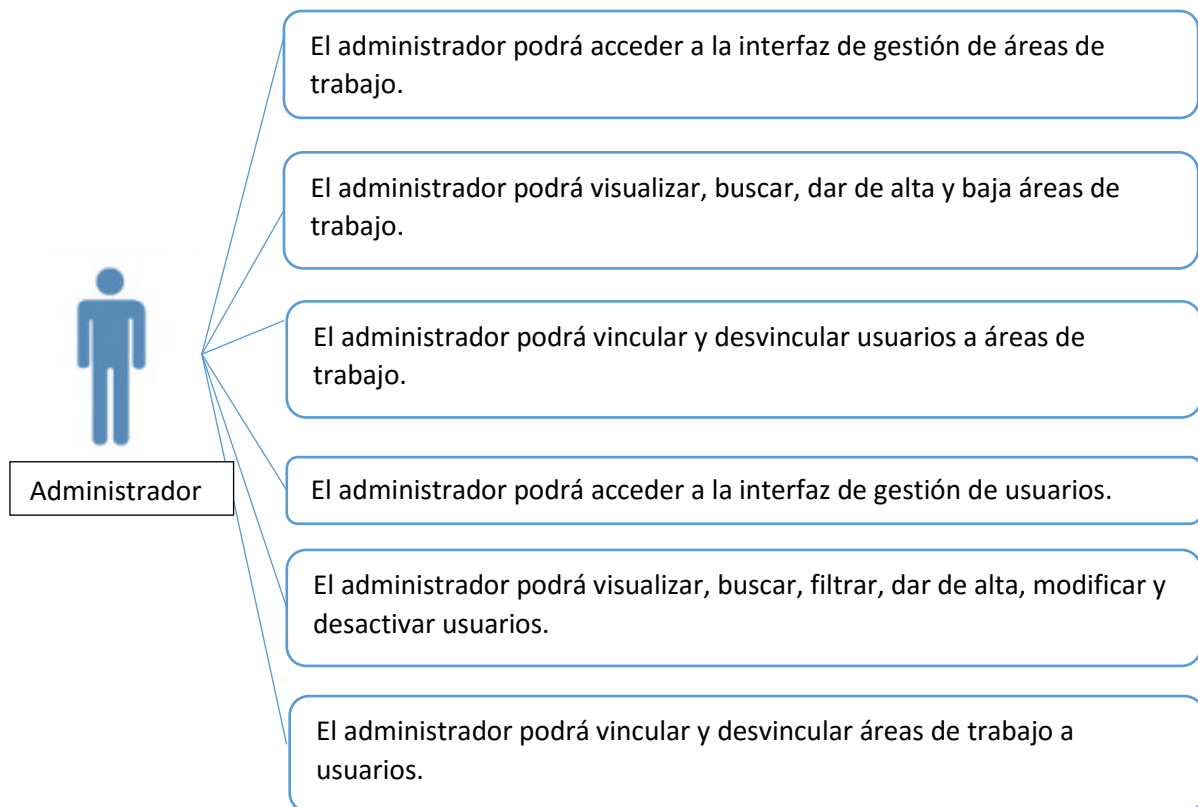


Ilustración 10 - Esquema de los casos de uso del administrador

A continuación, se presenta un esquema de los casos de uso a rasgos generales del sistema de la herramienta ETL:

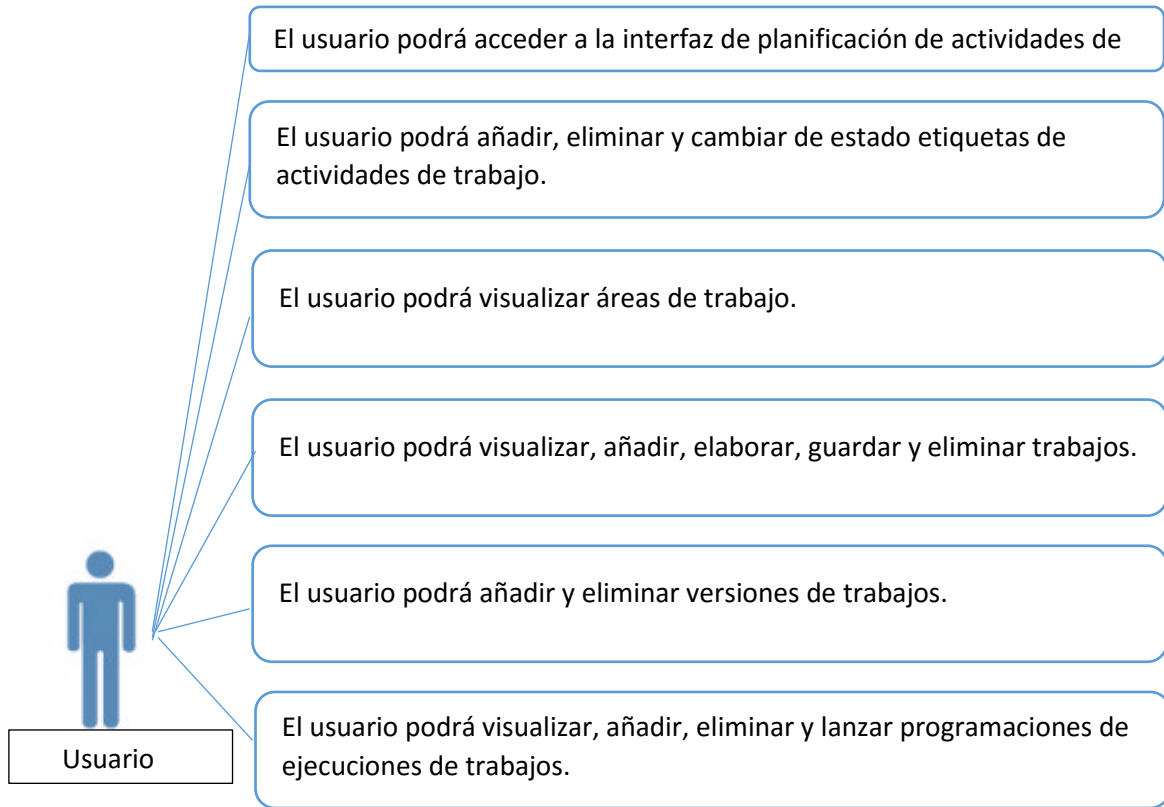


Ilustración 11 - Esquema de los casos de uso del usuario

Es importante señalar que, debido a la cantidad de funcionalidades que conlleva la aplicación, cada uno de los casos de uso que se van a presentar implicará varias tareas que el usuario podría realizar por separado.

Para exponer los casos de uso se empleará una tabla a modo de plantilla que contiene la siguiente información:

- **Identificador:** nomenclatura que identifica inequívocamente al caso de uso. Se define como CU-X, donde X es un número que representa el caso de uso.
- **Descripción:** breve descripción del caso de uso.
- **Actores:** los posibles roles de usuario que pueden intervenir en el caso de uso.
- **Precondiciones:** condiciones necesarias antes de realizar el caso de uso.
- **Pasos a seguir:** descripción guiada de los pasos que componen el caso de uso.

El modelo de plantilla a seguir, con los campos arriba mencionados, es el siguiente:

Identificador	Descripción
Actores	
Precondiciones	
Pasos a seguir	

Tabla 15 - Formato de las tablas de especificación de casos de uso

Identificador	CU-1	Descripción	El administrador gestiona las áreas de trabajo.
Actores	Administrador		
Precondiciones	Tener un usuario con perfil de administrador y haberse <i>logueado</i> en la aplicación.		
Pasos a seguir	<p>El administrador realiza los siguientes pasos:</p> <ol style="list-style-type: none"> 1. Seleccionar la pestaña “Áreas de trabajo”. 2. Hacer clic en el botón “Añadir área” para abrir el panel destinado a añadir un nuevo área de trabajo. 3. Insertar nombre del nuevo área en el campo etiquetado como “Nombre”. 4. Hacer clic en el botón “Añadir”. 5. Cerrar mensaje informativo. 6. Seleccionar nombre del área de trabajo a eliminar en el campo selector descrito como “Introduzca nombre area”. 7. Hacer clic en el botón con el símbolo de la papelera asociado al área de trabajo seleccionada. 8. Hacer clic en el botón “Eliminar” del panel de confirmación de borrado. 9. Seleccionar nombre del área de trabajo a visualizar en el campo selector descrito como “Introduzca nombre area”. 10. Hacer clic en el botón “Usuarios” asociado al área de trabajo para abrir panel de usuarios. 11. Seleccionar usuario por el nombre en el campo selector etiquetado como “Nombre de usuario”. 12. Hacer clic en el botón “Añadir”. 13. Hacer clic en el botón con el símbolo de la papelera asociado al usuario a desvincular del área de trabajo. 14. Hacer clic en el botón “Eliminar” del panel de confirmación de borrado. 		

Tabla 16 - Caso de uso CU-1

Identificador	CU-2	Descripción	El administrador gestiona los usuarios.
Actores	Administrador		
Precondiciones	Tener un usuario con perfil de administrador y haberse <i>logueado</i> en la aplicación.		

Pasos a seguir	<p>El administrador realiza los siguientes pasos:</p> <ol style="list-style-type: none"> 1. Seleccionar la pestaña “Usuarios”. 2. Hacer clic en el botón “Añadir usuario” para abrir el panel destinado a añadir un nuevo usuario. 3. Insertar nick, nombre, correo y rol del nuevo usuario en los campos etiquetados como “Nick”, “Nombre”, “Correo” y “Roles” respectivamente. 4. Hacer clic en el botón “Añadir”. 5. Cerrar mensaje informativo. 6. Seleccionar nombre del usuario a modificar en el campo selector descrito como “Introduzca nombre usuario”. 7. Hacer clic en el botón con el símbolo de edición asociado al usuario seleccionado para abrir el panel de edición. 8. Modificar nombre, roles y/o estado (activo o no activo) mediante los campos etiquetados como “Nombre”, “Roles”, “Estado” respectivamente. 9. Hacer clic en el botón “Editar”. 10. Cerrar mensaje informativo. 11. Seleccionar nombre del usuario a visualizar en el campo selector descrito como “Introduzca nombre usuario”. 12. Hacer clic en el botón “Áreas” asociado al usuario para abrir panel de áreas de trabajo. 13. Seleccionar área por el nombre en el campo selector etiquetado como “Área de trabajo”. 14. Hacer clic en el botón “Añadir”. 15. Hacer clic en el botón con el símbolo de la papelera asociado al área de trabajo a desvincular del usuario. 16. Hacer clic en el botón “Eliminar” del panel de confirmación de borrado. 17. Hacer clic en el botón desplegable con nombre “Filtro”. 18. Seleccionar “Administrador”, “Usuarios” o “Todos”. 19. Seleccionar “Activos” o “No activos”. 20. Visualizar el resultado del filtro en la tabla de usuarios.
-----------------------	---

Tabla 17 - Caso de uso CU-2

Identificador	CU-3	Descripción	El usuario planifica sus actividades de trabajo.
Actores	Usuario		
Precondiciones	Tener un usuario con perfil de usuario y haberse <i>logueado</i> en la aplicación.		

Pasos a seguir	<p>El usuario realiza los siguientes pasos:</p> <ol style="list-style-type: none"> 1. Seleccionar la pestaña “Tablero Kanban”. 2. Hacer clic sobre la cabecera de una actividad, sin soltar el ratón desplazar la etiqueta a otra columna de estado y soltar el ratón (arrastrar y soltar). 3. Hacer clic en el botón con el símbolo de una papelera asociado a una actividad para eliminarla. 4. Hacer clic en el botón con el símbolo más junto a la cabecera “Entrada” para abrir el panel para añadir nueva actividad. 5. Introducir descripción de la actividad en el campo “Contenido” del panel y pulsar el botón “Añadir”. 6. Hacer clic en el botón “Guardar”.
-----------------------	---

Tabla 18 - Caso de uso CU-3

Identificador	CU-4	Descripción	El usuario gestiona trabajos.
Actores	Usuario		
Precondiciones	Tener un usuario con perfil de usuario y haberse <i>logueado</i> en la aplicación. Estar vinculado a un área de usuario.		
Pasos a seguir	<p>El usuario realiza los siguientes pasos:</p> <ol style="list-style-type: none"> 1. Seleccionar la pestaña “Áreas de trabajo”. 2. Seleccionar área de trabajo del campo selector etiquetado como “Área seleccionada”. 3. Hacer clic en el botón “Seleccionar”. 4. Hacer clic en el botón “Añadir trabajo” en la vista del área de trabajo previamente seleccionada para abrir panel para añadir un nuevo trabajo. 5. Indicar nombre y descripción de trabajo en los campos habilitados para ello. 6. Hacer clic en botón “Añadir”. 7. Cerrar mensaje informativo. 8. Seleccionar trabajo del campo selector etiquetado como “Seleccione un trabajo de la lista”. 9. Seleccionar versión del trabajo existente del campo selector etiquetado como “Seleccione una versión” en la vista del trabajo cargada. 10. Hacer clic en el botón “Eliminar versión repositorio”. 11. Hacer clic en el botón “Eliminar” del panel de confirmación. 12. Cerrar mensaje informativo. 13. Hacer clic en el botón “Eliminar trabajo”. 14. Hacer clic en el botón “Eliminar” del panel de confirmación. 15. Cerrar mensaje informativo. 		

Tabla 19 - Caso de uso CU-4

Identificador	CU-5	Descripción	El usuario realiza un trabajo ETL.
Actores	Usuario		
Precondiciones	Tener un usuario con perfil de usuario y haberse <i>logueado</i> en la aplicación. Disponer de los ficheros de tipo CSV utilizados en la entrada del trabajo ETL y tener permisos de lectura sobre los mismos. Tener permisos de escritura sobre la ruta del fichero de tipo CSV utilizado como salida en el trabajo ETL. Los ficheros se encuentren en el servidor donde esté desplegada la aplicación.		

Pasos a seguir	<p>El usuario realiza los siguientes pasos:</p> <ol style="list-style-type: none"> 1. Seleccionar la pestaña “Áreas de trabajo”. 2. Seleccionar área de trabajo del campo selector etiquetado como “Área seleccionada”. 3. Hacer clic en el botón “Seleccionar”. 4. Seleccionar trabajo del campo selector etiquetado como “Seleccione un trabajo de la lista”. 5. Seleccionar versión del trabajo existente del campo selector etiquetado como “Seleccione una versión” en la vista del trabajo cargada. 6. Hacer clic en el botón “Abrir trabajo repositorio”. 7. Comprobar que se ha accedido al área de elaboración de procesos ETL del trabajo seleccionado. 8. Hacer clic en el botón “FicheroEntrada” del panel de componentes. 9. Hacer clic en el área de trabajo. 10. Hacer doble clic sobre el componente “FicheroEntrada” para abrir su panel de gestión. 11. Seleccionar pestaña “Mapeo campos” del panel. 12. Introducir nombre del campo y seleccionar tipo de campo. 13. Hacer clic en botón “Añadir”. 14. Seleccionar pestaña “Fichero”. 15. Introducir ruta completa del fichero CSV y campo separador de columna en los campos habilitados. 16. Hacer clic en botón “Aplicar”. 17. Hacer clic en botón “Cerrar” para cerrar panel de “FicheroEntrada”. 18. Hacer clic en el botón “FicheroEntrada” del panel de componentes. 19. Hacer clic en el área de trabajo. 20. Hacer doble clic sobre el componente “FicheroEntrada” para abrir su panel de gestión. 21. Seleccionar pestaña “Mapeo campos” del panel. 22. Introducir nombre del campo y seleccionar tipo de campo. 23. Hacer clic en botón “Añadir”. 24. Introducir el nombre de otro campo y seleccionar su tipo de campo. 25. Hacer clic en botón “Añadir”. 26. Seleccionar pestaña “Fichero”. 27. Introducir ruta completa del fichero CSV y campo separador de columna en los campos habilitados. 28. Hacer clic en botón “Aplicar”. 29. Hacer clic en botón “Cerrar” para cerrar panel de “FicheroEntrada”. 30. Hacer clic en el botón “Mapeo” del panel de componentes. 31. Hacer clic en el área de trabajo. 32. Hacer clic en el botón “Filtro” del panel de componentes.
-----------------------	---

33. Hacer clic en el área de trabajo.
34. Hacer clic en el botón "FicheroSalida" del panel de componentes.
35. Hacer clic en el área de trabajo.
36. Hacer doble clic sobre el primer componente "FicheroEntrada" para abrir su panel de gestión.
37. En la pestaña "Conector" del panel seleccionar el nombre del componente "Mapeo" del campo selector de conectores de salida.
38. Hacer clic en el botón "Aplicar".
39. Hacer doble clic sobre el segundo componente "FicheroEntrada" para abrir su panel de gestión.
40. En la pestaña "Conector" del panel seleccionar el nombre del componente "Mapeo" del campo selector de conectores de salida.
41. Hacer clic en el botón "Aplicar".
42. Hacer doble clic sobre el componente "Mapeo" para abrir su panel de gestión.
43. Seleccionar la pestaña "Mapeo campos" del panel.
44. Hacer clic en el botón con el símbolo de una papelera asociado al tercer registro que se corresponde con el segundo campo mapeado en el segundo componente "FicheroEntrada".
45. Seleccionar la pestaña "Relaciones".
46. Seleccionar en el primer selector con los campos del primer componente "FicheroEntrada" el primer campo.
47. Seleccionar en el segundo selector con los campos del segundo componente "FicheroEntrada" el primer campo.
48. Seleccionar el tipo de relación "Left join" o "Inner join".
49. Hacer clic en el botón con el símbolo de confirmación.
50. Seleccionar la pestaña "Conector".
51. Seleccionar el nombre del componente "Filtro" del campo selector de conectores de salida.
52. Hacer clic en el botón "Aplicar".
53. Hacer doble clic sobre el componente "Filtro" para abrir su panel de gestión.
54. Seleccionar la pestaña "Filtros".
55. Seleccionar el campo correspondiente al primer campo mapeado en el primer componente "FicheroEntrada" en el campo selector de campos.
56. Seleccionar si el campo va a ser igual o distinto a un valor en el segundo campo selector.
57. Escribir el valor a comparar con el del campo indicado en el tercer campo descrito como "filtro".
58. Hacer clic en el botón con el símbolo de confirmación.
59. Seleccionar la pestaña "Conector".
60. Seleccionar el nombre del componente "FicheroSalida" del campo selector de conectores de salida.

	61. Hacer clic en el botón “Aplicar”. 62. Hacer doble clic sobre el componente “FicheroSalida” para abrir su panel de gestión. 63. Seleccionar la pestaña “Fichero”. 64. Indicar la ruta del fichero CSV de salida en el campo etiquetado como “Ruta fichero”. 65. Hacer clic en el botón “Aplicar”. 66. Hacer clic en el botón “Guardar”. 67. Cerrar mensaje informativo. 68. Hacer clic en el botón “Repositorio” para abrir panel de repositorio del trabajo. 69. Seleccionar número de versión del campo selector etiquetado como “Versión”. 70. Escribir la descripción de la versión en el campo etiquetado como “Descripción”. 71. Hacer clic en el botón “Subir”. 13. Hacer clic en el botón “Confirmar” del panel de confirmación. 72. Cerrar mensaje informativo. 73. Pulsar botón “Salir trabajo” situado en la cabecera.
--	---

Tabla 20 - Caso de uso CU-5

Identificador	CU-6	Descripción	El usuario gestiona la programación de trabajos.
Actores	Usuario		
Precondiciones	Tener un usuario con perfil de usuario y haberse <i>logueado</i> en la aplicación. Disponer de los ficheros de tipo CSV utilizados en la entrada de los trabajos ETL programados y tener permisos de lectura sobre los mismos. Tener permisos de escritura sobre la ruta de los ficheros de tipo CSV utilizados como salida en los trabajos ETL programados. Los ficheros se encuentren en el servidor donde esté desplegada la aplicación.		
Pasos a seguir	El usuario realiza los siguientes pasos: 1. Seleccionar la pestaña “Programación”. 2. Seleccionar trabajo del campo selector “Nombre trabajo”. 3. Seleccionar versión de trabajo del campo selector “Versión”. 4. Escribir el horario de programación en el campo de texto “Programación”. 5. Hacer clic en botón con el símbolo de confirmación. 6. Cerrar el mensaje de confirmación de alta. 7. Hacer clic en el botón con el símbolo <i>play</i> asociado al trabajo a ejecutar. 8. Cerrar el mensaje de ejecución de trabajo. 9. Consultar los campos “Última ejecución”, “Estado” y “Descripción” del trabajo ejecutado. 10. Hacer clic en el botón con el símbolo de la papelera asociado a la programación a eliminar.		

Tabla 21 - Caso de uso CU-6

Identificador	CU-7	Descripción	El usuario reinicia clave de usuario y se <i>loguea</i> .
Actores	Administrador, Usuario		
Precondiciones	Tener una cuenta de usuario. Activar opción de “Acceso de aplicaciones menos seguras” en la configuración del correo asociado a la cuenta de usuario.		
Pasos a seguir	<p>El usuario realiza los siguientes pasos:</p> <ol style="list-style-type: none"> 1. Hacer clic en el enlace “Reinicio de clave de usuario”. 2. Introducir correo en el campo etiquetado como “Correo”. 3. Hacer clic en el botón “Enviar solicitud”. 4. Hacer clic en el enlace del correo recibido con el asunto “[Open Web ETL] Reinicio de clave”. 5. En la página abierta desde el enlace, introducir la nueva clave en el campo etiquetado como “Clave”. 6. Hacer clic en el botón “Cambiar clave”. 7. Esperar mensaje “Cambio de clave correcto”. 8. Hacer clic en el enlace de la cabecera “Acceder”. 9. En la página de acceso a la que se ha llegado introducir el nick y la clave de usuario en los campos correspondientes. 10. Hacer clic en el botón “Acceder”. 		

Tabla 22 - Caso de uso CU-7

4.2.1.3. Trazabilidad

En este punto se realizará un estudio para detectar inconsistencias, ambigüedad, duplicidad o escasez de información. Dicho estudio consistirá en el análisis de los requisitos y de los casos de uso, asociándolos para identificar sus funcionalidades y comportamientos comunes.

Para ello se realizará una matriz de trazabilidad entre los distintos casos de uso y los requisitos descritos. De esta manera, el usuario se asegurará de que los requisitos especificados en el catálogo de requisitos son válidos, consistentes y completos, y de que al menos existe un requisito por cada caso de uso definido anteriormente.

A continuación se muestra la matriz de trazabilidad de los Requisitos de Software Funcionales y los Casos de Uso:

CASOS REQUISITOS	CU-1	CU-2	CU-3	CU-4	CU-5	CU-6	CU-7
RRG-01							x
RRG-02							x
RRG-03							x
RFAA-04	x						
RRAA-05	x						
RRAA-06	x						
RRAA-07	x						

CASOS REQUISITOS	CU-1	CU-2	CU-3	CU-4	CU-5	CU-6	CU-7
RRAA-08	x						
RRAA-09	x						
RRAA-10	x						
RFAA-11	x						
RRAA-12	x						
RFAA-13	x						
RRAA-14	x						
RRAA-15	x						
RFAA-16	x						
RRAA-17	x						
RRAA-18	x						
RFAA-19	x						
RRAA-20	x						
RRAA-21	x						
RRAA-22	x						
RRAA-23	x						
RFAA-24	x						
RRAA-25	x						
RFAA-26	x						
RRAA-27	x						
RRAA-28	x						
RFAU-29		x					
RRAU-30		x					
RRAU-31		x					
RRAU-32		x					
RRAU-33		x					
RRAU-34		x					
RRAU-35		x					
RRAU-36		x					
RFAU-37		x					
RRAU-38		x					
RRAU-39		x					
RRAU-40		x					
RRAU-41		x					
RFAU-42		x					
RRAU-43		x					
RRAU-44		x					
RFAU-45		x					
RRAU-46		x					
RFAU-47		x					
RRAU-48		x					
RRAU-49		x					
RRAU-50		x					
RRAU-51		x					
RFAU-52		x					
RRAU-53		x					
RFAU-54		x					
RRAU-55		x					
RRAU-56		x					

CASOS REQUISITOS	CU-1	CU-2	CU-3	CU-4	CU-5	CU-6	CU-7
RFTP-57			x				
RRTP-58			x				
RRTP-59			x				
RRTP-60			x				
RFTP-61			x				
RRTP-62			x				
RFTP-63			x				
RRTP-64			x				
RFTP-65			x				
RRTP-66			x				
RFTP-67			x				
RRTP-68			x				
RFTP-69			x				
RRTP-70			x				
RFTE-71						x	
RRTE-72						x	
RFTE-73						x	
RRTE-74						x	
RRTE-75						x	
RFTE-76						x	
RRTE-77						x	
RFTE-78						x	
RRTE-79						x	
RRTE-80						x	
RFTG-84				x	x		
RRTG-85				x	x		
RRTG-86				x	x		
RRTG-87				x	x		
RFTG-88				x	x		
RFTG-89				x	x		
RRTG-90				x	x		
RFTG-91				x			
RRTG-92				x			
RRTG-93				x			
RRTG-94				x			
RRTG-95				x			
RFTG-96				x	x		
RRTG-97				x	x		
RFTG-98				x			
RRTG-99				x			
RRTG-100				x			
RFTG-101				x	x		
RRTG-102				x	x		
RFTG-103				x	x		
RRTG-104				x	x		
RFTG-105				x			
RRTG-106				x			
RFTG-107					x		
RRTG-108					x		

CASOS REQUISITOS	CU-1	CU-2	CU-3	CU-4	CU-5	CU-6	CU-7
RFTG-109					x		
RRTG-110					x		
RFTG-111					x		
RRTG-112					x		
RRTG-113					x		
RRTG-114					x		
RRTG-115					x		
RRTG-116					x		
RRTG-117					x		
RRTG-118					x		
RRTG-119					x		
RRTG-120					x		
RRTG-121					x		
RRTG-122					x		
RRTG-123					x		
RRTG-124					x		
RRTG-125					x		
RRTG-126					x		
RRTG-127					x		
RRTG-128					x		
RRTG-129					x		
RRTG-130					x		
RRTG-131					x		
RRTG-132					x		
RRTG-133					x		
RRTG-134					x		
RRTG-135					x		
RRTG-136					x		
RRTG-137					x		
RRTG-138					x		
RRTG-139					x		
RRTG-140					x		
RRTG-141					x		
RRTG-142					x		
RRTG-143					x		
RRTG-144					x		
RRTG-145					x		
RRTG-146					x		
RRTG-147					x		
RRTG-148					x		
RRTG-149					x		

Tabla 23 - Matriz de trazabilidad de los Requisitos de SW Funcionales y los Casos de Uso

Como se puede observar todos los requisitos tienen al menos un caso de uso y viceversa.

4.2.2. Diseño

En la etapa de diseño se llevan a cabo cuatro propiedades diferentes del software: la estructura de los datos, la arquitectura de las aplicaciones, la estructura interna de los programas y las interfaces.

En esta etapa se traducen los requisitos en una representación del software de manera que pueda saberse la arquitectura, funcionalidad e inclusive la calidad de éste antes de poner en marcha la codificación.

4.2.2.1. Contexto del sistema

Es necesario disponer de una visión global del sistema antes de iniciar el diseño de la herramienta para de esta forma tener presente dónde se encuentra enmarcada, los sistemas que interactúan dentro de ese marco y sus límites.

La finalidad de la aplicación es ofrecer una interfaz con vistas simples y elegantes que permita a un usuario elaborar trabajos de procesos ETL mediante programación por componentes y gestionarlos de una manera muy visual, sencilla e intuitiva. Se recuerda que los procesos ETL que soporta la herramienta se centran en trabajos sobre ficheros de tipo CSV:

- Lectura de ficheros de tipo CSV de los que extraer los datos.
- Mapeo de campos de ficheros de tipo CSV pudiendo relacionar dos ficheros a través de sus campos, relación de tipo *left join* o *inner join*.
- Filtrado de filas del flujo de datos mediante la comparación de campos mapeados con valores proporcionados.
- Carga de datos en ficheros de tipo CSV.

El usuario, a través de la interfaz de la aplicación, es el encargado de proporcionar los datos que alimentan la base de datos con la que trabaja el sistema. De igual forma, es el propio usuario el que se ocupa de poner los ficheros de entrada requeridos por un proceso ETL antes de su ejecución.

Estos ficheros de entrada deben ser ubicados por el usuario en el servidor donde se encuentre desplegada la aplicación en la misma ruta indicada en el trabajo del proceso ETL. La idea es que el usuario no tenga que añadir los ficheros directamente en el servidor de la aplicación sino que lo haga a través de unidades remotas montadas tanto en el equipo del cliente como en este servidor. Sin embargo, el usuario es libre de mover o copiar los ficheros de entrada directamente a este servidor.

Es importante procurar permisos de lectura a la herramienta sobre los ficheros de entrada y garantizar que los mismos son de tipo CSV. Por otro lado, los procesos ETL generados con la herramienta cargan el flujo de datos que transforman en ficheros de salida de tipo CSV cuyas rutas son especificadas por el usuario desde la aplicación. Por consiguiente, también es importante garantizar que la herramienta tiene permisos de escritura sobre el directorio en el que se crearán estos ficheros de salida.

Por último, se presenta la arquitectura de la aplicación en la siguiente ilustración:

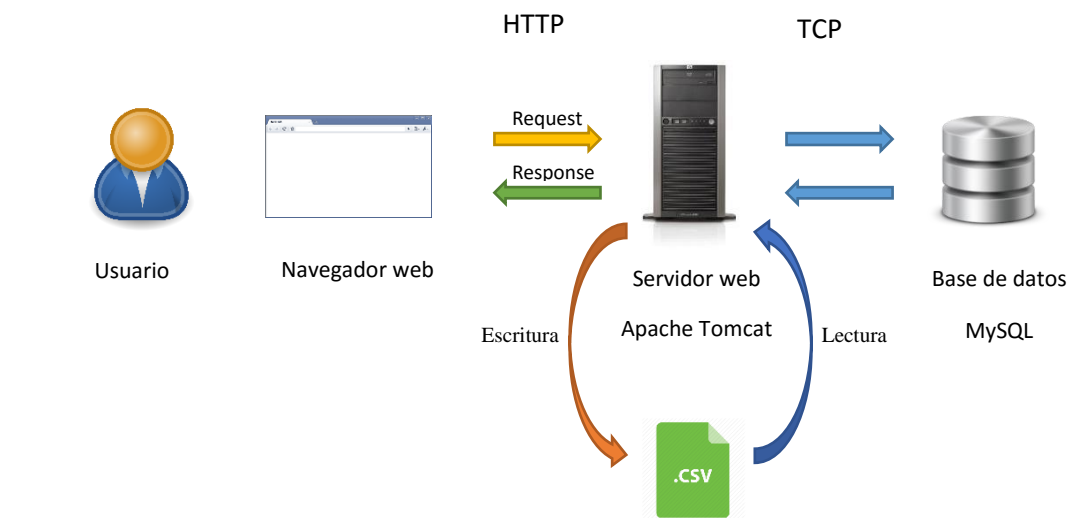


Ilustración 12- Arquitectura de la aplicación

La construcción de la base de datos de la aplicación forma parte de este proyecto y su estructura será detallada en el siguiente apartado del diseño. Hay que señalar que la base de datos no tiene por qué estar alojada en el mismo servidor en el que se encuentre desplegada la aplicación web.

4.2.2.2. Diseño del sistema

Es necesario tener en cuenta las herramientas para el desarrollo del sistema, que en este caso tendrán como base Java para la parte del servidor. Se utilizará como framework principal Spring Framework puesto que cambia las responsabilidades y en lugar de que el propio desarrollador sea el encargado de generar los objetos de cada uno de los frameworks es Spring basándose en ficheros xml o anotaciones (para este proyecto se utilizarán anotaciones) el responsable de construir todos los objetos que la aplicación utilizará. De esta forma, al convertirse Spring en el encargado de inicializar todos los objetos de los distintos frameworks, es además el responsable de asegurar que se integran de la manera correcta.

La aplicación tendrá un acceso restringido, existiendo dos tipos de roles: administrador y usuario. Un usuario con un determinado rol sólo podrá acceder a cierta información del sistema. Con el fin de controlar este acceso a la aplicación y el sistema de roles se ha usado Spring Security, que es un módulo del framework Spring que permite gestionar la seguridad de aplicaciones Java. Se ha elegido esta tecnología para controlar la seguridad porque es capaz de gestionar seguridad en varios niveles, es decir, URLs que se piden al servidor, acceso a métodos y clases Java y acceso a instancias específicas de las clases. Además, permitirá separar la lógica de la aplicación del control de la seguridad, haciendo uso de filtros para las solicitudes al servidor de aplicaciones o temas para la seguridad en clases y métodos. Permitirá, por medio de anotaciones en los métodos, evitar que un usuario de la aplicación que no tenga el rol definido en el propio método sea rechazado.

Spring web MVC framework es el otro módulo del framework Spring que se empleará. Este framework ofrece la arquitectura modelo-vista-controlador y componentes preparados para ser utilizados para desarrollar aplicaciones web flexibles y débilmente acopladas. El patrón MVC se traduce en la separación de los distintos aspectos de la aplicación, es decir, la lógica de entrada,

la lógica de negocio y la lógica de la interfaz de usuario, mientras que ofrece un acoplamiento débil entre estos elementos.

- Modelo: encapsula los datos de la aplicación.
- Vista: muestra los datos del modelo, en esta aplicación se generará una salida HTML que el navegador web del cliente podrá interpretar.
- Controlador: se encarga de procesar las peticiones del usuario y la construcción del modelo apropiado, pasándolo a la vista para la representación.

Se empleará una cuarta capa denominada Servicio que será la encargada de leer y escribir de la base de datos de la aplicación, enviándole los resultados al Controlador. Es decir, el Controlador es el que pide al Servicio que haga una determinada acción sobre el medio físico en el que se guardan los datos.

Un aspecto importante que se ha tenido en consideración para decantarse por Spring web MVC framework ha sido que, a partir de Spring 3.0, el mecanismo `@Controller` permite crear aplicaciones Web RESTful mediante anotaciones. De esta forma, los métodos del controlador serán consumidos como servicios web, aspecto importante puesto que la aplicación web diseñada será una SPA.

Una aplicación de página única (SPA) es una aplicación web que puede ubicarse en una única página con la intención de que el usuario tenga una experiencia más fluida como si se tratara de una aplicación de escritorio. Todos los códigos de HTML, JavaScript y CSS se cargan de una sola vez o los recursos necesarios se cargan dinámicamente según lo requiera la página y se van añadiendo como resultado de las acciones del usuario. No se precia que la página sea recargada. Existen distintos frameworks de JavaScript que han adquirido los principios de SPA. En este caso, se utilizará Angularjs.

Angularjs es un framework JavaScript de desarrollo de aplicaciones web en el lado del cliente que trae consigo el módulo *ngRoute* que ayuda a convertir una aplicación en SPA. Este módulo permite tener una aplicación de una única página, pero es capaz de representar URLs distintas, simulando una navegación a través de la aplicación. Para ello, hay que especificar una URL que se corresponderá con un enlace interno, la plantilla HTML a mostrar y el controlador con las funciones JavaScript que tendrá asociadas. En la única página real que se tenga se deberá especificar con una etiqueta especial de Angularjs el lugar donde la plantilla HTML se mostrará al acceder a la ruta correspondiente al enlace interno definido anteriormente.

Las comunicaciones dinámicas con el servidor web se harán utilizando el servicio de Angularjs llamado *\$http* haciendo uso de las funciones que ofrece, principalmente *get()* y *post()*. La técnica que utiliza es Ajax, que permite mantener la comunicación asíncrona con el servidor en segundo plano. Así, se pueden ejecutar cambios sobre las páginas sin tener que recargarlas aumentando la interactividad, velocidad y usabilidad de la aplicación.

\$http llamará a los métodos de los controladores a través de sus URLs de acceso, puesto que como se ha explicado anteriormente, van a ser servicios Web. El intercambio de datos se hará utilizando el formato JSON y para esto se empleará la librería Jackson.

Jackson es una librería de utilidad de Java que ayuda a simplificar el trabajo de convertir un objeto Java en una cadena de texto con su representación JSON y convertir una cadena de texto con una representación de JSON de un objeto en un objeto de Java.

Por otra parte, para conseguir un diseño profesional y fácil de implementar se utilizarán Bootstrap y jQueryUI, que son dos frameworks para el diseño de aplicaciones web. Contienen plantillas de diseño con formularios, botones, cuadros, menús de navegación y otros elementos de diseño basado en HTML y CSS, así como , extensiones de JavaScript opcionales adicionales.

Además, se emplearán las librerías de JavaScript Angularjs y jQuery para principalmente poder simplificar la forma de interactuar con los documentos HTML, manipular el árbol DOM y manejar eventos.

Por último, se ha comentado anteriormente que existe una capa de servicio que se ocupa de extraer y recuperar datos del esquema de base de datos de la aplicación. El diseño de este esquema forma parte de este proyecto y su modelo Entidad/Relación se presenta a continuación:

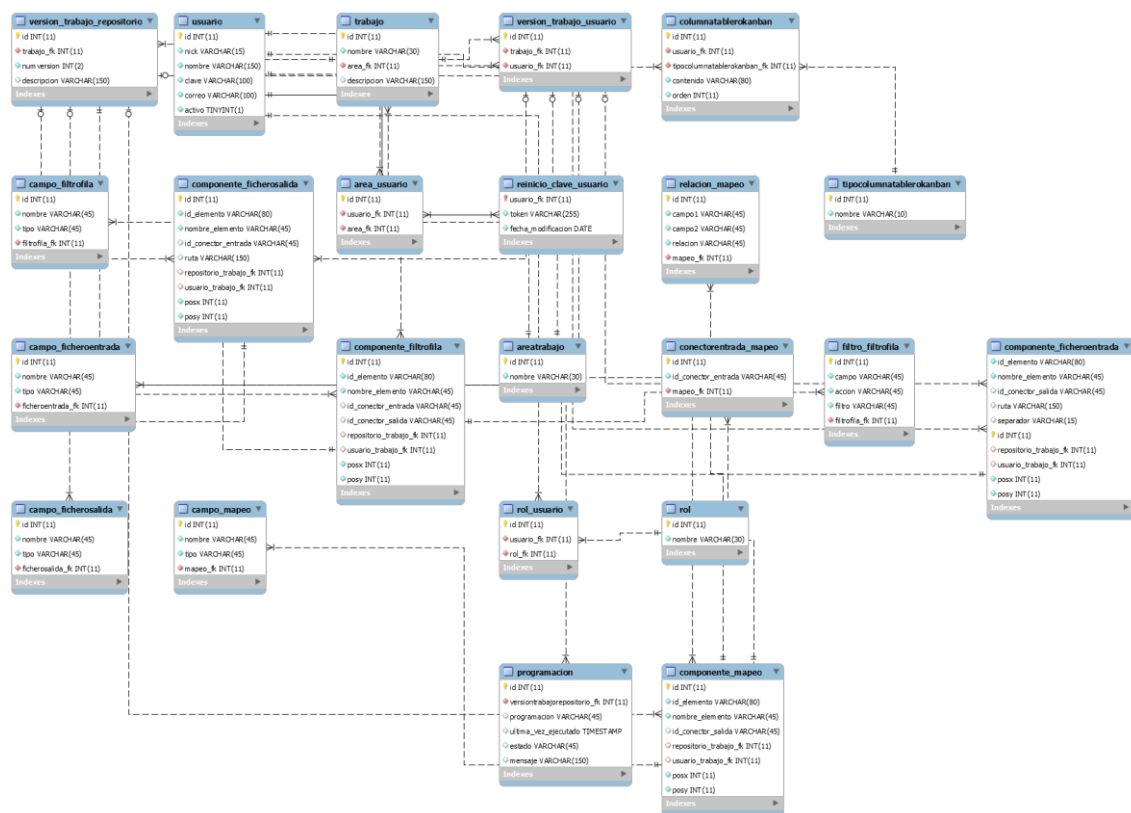


Ilustración 13 - Modelo Entidad/Relación de la base de datos de la aplicación

En el modelo entidad/relación del esquema de base de datos de la aplicación se han detallado las tablas que lo conforman y las relaciones entre éstas. Asimismo, han quedado reflejados los campos de las tablas y el tipo de datos de cada uno.

4.2.2.3. Diseño detallado

En este punto se detallará el diseño final de la aplicación, sabiendo que los cambios realizados en el diseño durante el desarrollo han sido prácticamente inexistentes y de escasa relevancia como puede ser la manera de expresar el contenido de las ventanas modales informativas.

Estos diseños han sido elaborados mediante capturas de pantalla de páginas HTML creadas desde cero empleando las plantillas de diseño con formularios, botones, cuadros, menús de navegación y otros elementos de diseño basado en HTML y CSS que ofrece Bootstrap. Este framework para el diseño de aplicaciones web es el mismo que el empleado en la herramienta desarrollada, por lo que el diseño aquí mostrado coincide fielmente con el del resultado final.

Es importante señalar que los botones, campos de texto, tablas y demás elementos que conforman las interfaces están diseñados para que se redimensionen y ajusten al tamaño de la pantalla de la ventana del navegador desde la que el usuario accede a la aplicación. Además, para un mejor aprendizaje y experiencia del usuario con la aplicación, los botones tienen asociados símbolos que identifican de manera visual las operaciones que ejecutan, por ejemplo, los botones que ejecutan acciones de borrado tienen el símbolo de una papelera. Asimismo, el color de un botón indica la operación que ejecuta éste, de esta forma un botón de color rojo ejecutará acciones de borrado y un botón verde ejecutará acciones para añadir elementos o guardar información.

Por otro lado, las ventanas modales empleadas para añadir, modificar o eliminar elementos disponen de dos botones en el pie. El primer botón estará siempre alineado a la izquierda y será el que ejecute la operación y el segundo estará alineado a la derecha y será el que cierre la ventana sin realizar la operación. Mediante esta disposición, se pretende evitar que el usuario pulse uno de los botones por accidente por estar demasiado juntos. Del mismo modo, sabrá que el botón de la derecha siempre será el que ejecute la operación y el de la izquierda el que permita salir sin realizarla.

Otro aspecto que contribuye a que el usuario tenga una buena experiencia con la aplicación es que la navegación es fluida, ya que el acceso a las distintas vistas no requiere recargar la página en la que se encuentran.

Existe un área específica para la administración y otra para la gestión de los trabajos que sólo tienen en común la vista de acceso a la aplicación. Esta vista cuenta con un contenedor centrado en la pantalla cuya cabecera contiene el título de la aplicación y una breve descripción de su cometido y cuyo cuerpo cuenta con campos para introducir el *nick* y la clave del usuario y un botón para acceder a cada una de las áreas mencionadas anteriormente según se acceda con perfil de Administrador o de Usuario respectivamente.



Ilustración 14 - Página de acceso

Se ha acordado con el cliente no detallar el diseño del área de administración ni el diseño de reinicio de contraseña, por lo que se procederá a explicar el área de trabajo que supone el núcleo de las funcionalidades que ofrece el sistema.

- **Área de Trabajo**

Incluye una barra de navegación en la cabecera que muestra alineados a la izquierda y de izquierda a derecha el título de la aplicación y los tres enlaces para acceder a las distintas secciones principales del área de trabajo: 'Tablero Kanban', 'Áreas de Trabajo' y 'Programación'. Alineado a la izquierda se muestra el enlace 'Salir' para finalizar la sesión de usuario y volver a la página de acceso. Esta barra de navegación es visible en todo momento dentro del área de trabajo, independientemente de la sección en la que se encuentre el usuario de la aplicación.

A continuación, se va a describir el diseño de cada una de las tres secciones principales del área de trabajo mencionadas anteriormente que estarán enmarcadas debajo de la barra de navegación.

- **Tablero Kanban**

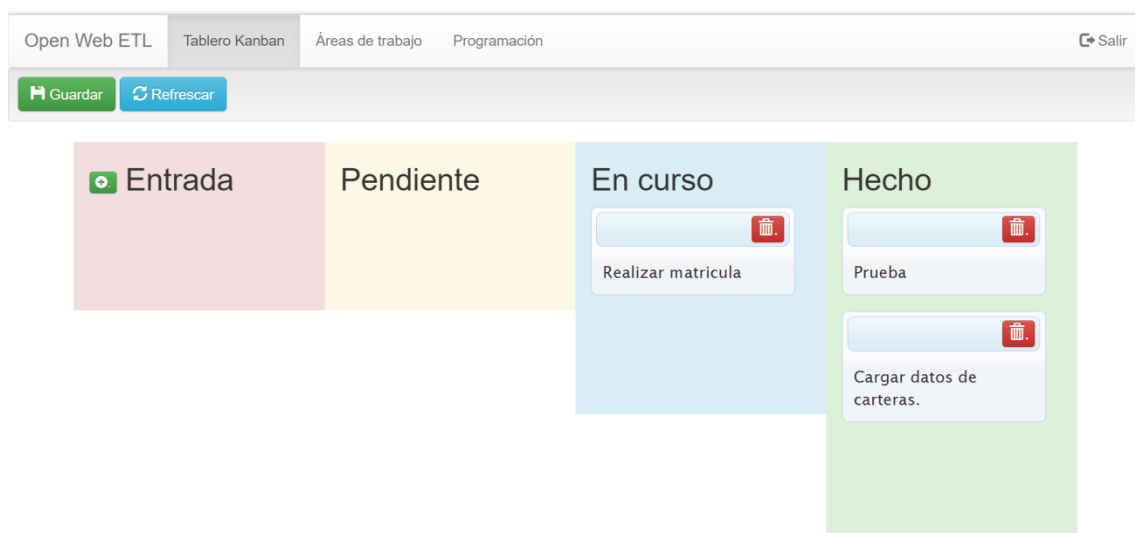


Ilustración 15 - Diseño tablero Kanban

Existe una barra de acciones situada justo debajo de la barra de navegación que contiene dos botones alineados a la izquierda. El primer botón con nombre 'Guardar' es de color verde y contiene el símbolo de un disquete para que el usuario identifique rápidamente la acción que ejecuta este botón. Cuando el usuario hace clic en este botón, el tablero de planificación de actividades de trabajo (Kanban) que refleja la vista se guardará en base de datos. El segundo botón con nombre 'Refrescar' es de color azul y contiene el símbolo de recarga para que el usuario identifique rápidamente la acción que ejecuta este botón. Cuando el usuario hace clic en este botón, la vista refleja el tablero tal y como se encuentra en base de datos.

Cada columna representa el estado en el que se encuentra una actividad de trabajo contenida en su interior. Cada una tiene un color diferente y un título con el nombre de su estado para que el usuario pueda distinguirlas sin ninguna dificultad. La primera columna se corresponde con el estado 'Entrada' y es de color rojo. La segunda columna se corresponde con el estado

‘Pendiente’ y es del color amarillo. La tercera columna se corresponde con el estado ‘En curso’ y es de color azul. Por último, la cuarta columna se corresponde con el estado ‘Hecho’ y es de color verde.

En la primera columna que se corresponde con el estado ‘Entrada’ está situado un botón junto al título ‘Entrada’. Este botón es de color verde y contiene el símbolo de suma para que el usuario identifique rápidamente la acción que ejecuta este botón. Cuando el usuario hace clic en el mismo, se abre una ventana modal desde la que añadir una actividad de trabajo.

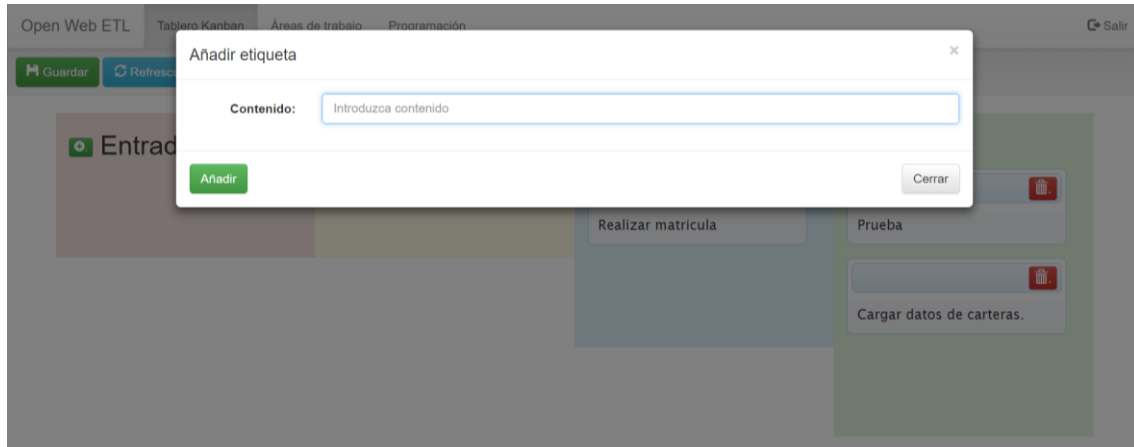


Ilustración 16 - Diseño añadir actividad de trabajo

Esta ventana modal incluye el título ‘Añadir etiqueta’ en su cabecera para que el usuario conozca su cometido. En el cuerpo incluye un campo de texto para insertar la descripción de la actividad etiquetado como ‘Contenido’ para su correcta identificación. Por último, en el pie de la ventana incluye dos botones. Alineado a la izquierda se encuentra el botón ‘Añadir’ de color verde para añadir la nueva actividad de trabajo y cerrar la ventana. Alineado a la derecha se encuentra el botón ‘Cerrar’ de color gris que cierra la ventana sin efectuar ninguna operación.

Cada actividad de trabajo está representada en la vista por un contenedor con cabecera y cuerpo. En la cabecera aparece alineado a la derecha un botón rojo con el símbolo de una papelera que borra de la vista el contenedor y ajusta el resto de contenedores de la columna para ocupar su posición. Por su parte, el cuerpo contiene la descripción de la actividad de trabajo.

Se ha procurado que el cambio de estados de las actividades fuera lo más sencillo e intuitivo posible, para lograr este objetivo se emplea la operación de arrastrar y soltar de manera que cada columna de estado acepte un elemento contenedor de actividades de trabajo. Esta acción se realiza haciendo clic en la cabecera del contenedor de actividad y, sin levantar el dedo del ratón, desplazar dicho contenedor hasta la columna de estado deseada. Es importante la posición de la columna en la que el usuario suelte el contenedor, puesto que si desea incluirlo entre dos contenedores ya existentes en esa columna deberá soltar el contenedor estando el cursor del ratón entre ambos contenedores, siguiendo el mismo procedimiento se logra incluir un contenedor antes o después de una actividad.

○ Áreas de trabajo

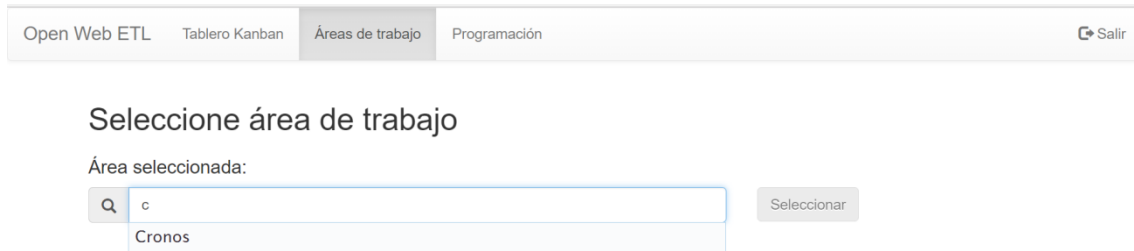


Ilustración 17 - Diseño selección área de trabajo

La sección para elaborar y gestionar trabajos se encuentra justo debajo de la barra de navegación y comienza con el título 'Programación de procesos' que ayuda al usuario a saber en qué punto de la aplicación se encuentra. Incluye un campo autocompletar debajo de la etiqueta 'Área seleccionada:' y precedido por un icono con el símbolo de una lupa para que el usuario identifique claramente que dicho campo es de búsqueda. La búsqueda de áreas de trabajo vinculadas al usuario de la aplicación es intuitiva reduciéndose el número de opciones de búsqueda a medida que se van insertando caracteres. Una vez encontrado el nombre del área de trabajo buscada, el usuario debe hacer clic sobre el mismo y de esta manera aparecerá el nombre del área seleccionada a continuación de la etiqueta 'Área seleccionada:'. En este momento, el botón 'Seleccionar' que se encuentra a continuación del campo de búsqueda se activará y, haciendo clic sobre el mismo, el usuario accederá a la zona de gestión de trabajos del área de trabajo seleccionada.

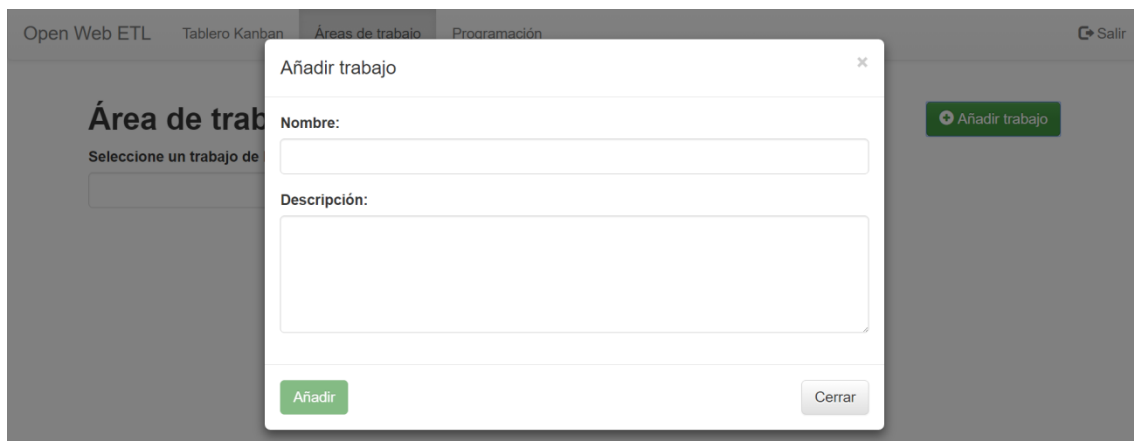


Ilustración 18 - Diseño incluir trabajo

Llegados a este punto, la sección situada debajo del menú de navegación ha cambiado y en su lugar el usuario observa una nueva sección con el título 'Área de trabajo: [nombre del área de trabajo a la que ha accedido el usuario]'. En la parte superior izquierda de esta sección se ubica un botón de color verde con el símbolo de suma y con el nombre 'Añadir trabajo' para que el usuario identifique claramente la acción que ejecuta este botón. Cuando el usuario hace clic en este botón, se abre una ventana modal cuya cabecera contiene el título 'Añadir trabajo' para que el usuario sepa que esta ventana se utiliza para añadir un trabajo. El cuerpo de esta ventana incluye un campo de texto identificado con la etiqueta 'Nombre' para que el usuario indique el nombre que tendrá el nuevo trabajo y un área de texto identificado con la etiqueta 'Descripción'

para que el usuario indique la descripción que tendrá el nuevo trabajo. Por último, en el pie de la ventana incluye dos botones. Alineado a la izquierda se encuentra el botón 'Añadir' de color verde para añadir el nuevo trabajo y cerrar la ventana. Alineado a la derecha se encuentra el botón 'Cerrar' de color gris que cierra la ventana sin efectuar ninguna operación.

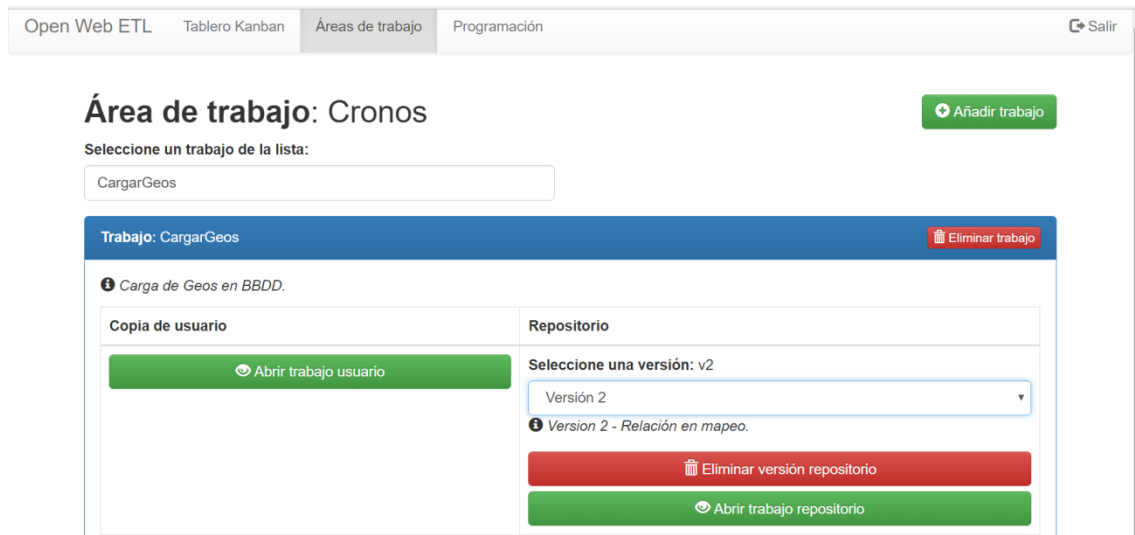


Ilustración 19 - Diseño área de trabajo

Se Incluye un campo autocompletar debajo del título 'Área de trabajo' identificado con la etiqueta 'Seleccione un trabajo de la lista:' para que el usuario identifique claramente que dicho campo es de búsqueda de trabajos. La búsqueda de trabajos pertenecientes al área de trabajo en el que se encuentra el usuario de la aplicación es intuitiva reduciéndose el número de opciones de búsqueda a medida que se van insertando caracteres. Una vez encontrado el nombre del trabajo buscado, el usuario debe hacer clic sobre el mismo y de esta manera aparecerá debajo de este campo de búsqueda un contenedor claramente identificable por el color azul de su cabecera que contendrá la información de ese trabajo seleccionado.

El contenedor de trabajo incluye en su cabecera el nombre del trabajo alineado a la izquierda y un botón de color rojo con nombre 'Eliminar trabajo' y el símbolo de una papelera para que el usuario identifique claramente la acción que ejecuta este botón. Cuando el usuario hace clic en este botón, se abre una ventana modal de confirmación para que pueda eliminar el trabajo. Tras esta acción, el contenedor del trabajo se elimina de la lista y en el campo autocompletar de trabajos deja de aparecer el nombre del mismo.

El cuerpo del contenedor de trabajo comienza con un icono de información seguido de la descripción del trabajo cargado. Este icono ayuda al usuario a identificar el texto que le precede como informativo. Debajo de la descripción del trabajo aparecen dos columnas: la de la izquierda para acceder al área para elaborar procesos ETL con la copia del trabajo del usuario con el título 'Copia de usuario' y la de la derecha para acceder a ese mismo área pero desde una versión del repositorio del trabajo con el título 'Repositorio'.

En la columna de la copia de usuario aparece un botón de color verde con el símbolo de un ojo abierto y de nombre 'Abrir trabajo usuario' para que el usuario identifique claramente la acción

que ejecuta este botón. Cuando el usuario hace clic en este botón, accede al área para la elaboración de procesos ETL con la copia guardada del usuario, si existiera.

En la columna del repositorio aparece un campo selector cuyas opciones son las versiones del trabajo guardadas en el repositorio. Haciendo clic en una de las versiones aparecerá encima de este campo selector el número de versión seleccionada y debajo de este campo un icono de información seguido de la descripción de la versión. Debajo de esta descripción se habilitará un botón de color rojo con el símbolo de una papelera y de nombre 'Eliminar versión repositorio'. Cuando el usuario hace clic en este botón, aparecerá una ventana modal de confirmación para eliminar la versión. Tras ejecutar esta acción, se eliminará el número de versión y la descripción de la vista y el campo selector no mostrará la versión eliminada.

Debajo del botón 'Eliminar versión repositorio' se habilitará otro botón tras haber seleccionado una versión. Este botón es de color verde con el símbolo de un ojo abierto y de nombre 'Abrir trabajo repositorio' para que el usuario identifique claramente la acción que ejecuta este botón. Cuando el usuario hace clic en este botón, accede al área para la elaboración de procesos ETL con la copia de la versión de repositorio seleccionada.

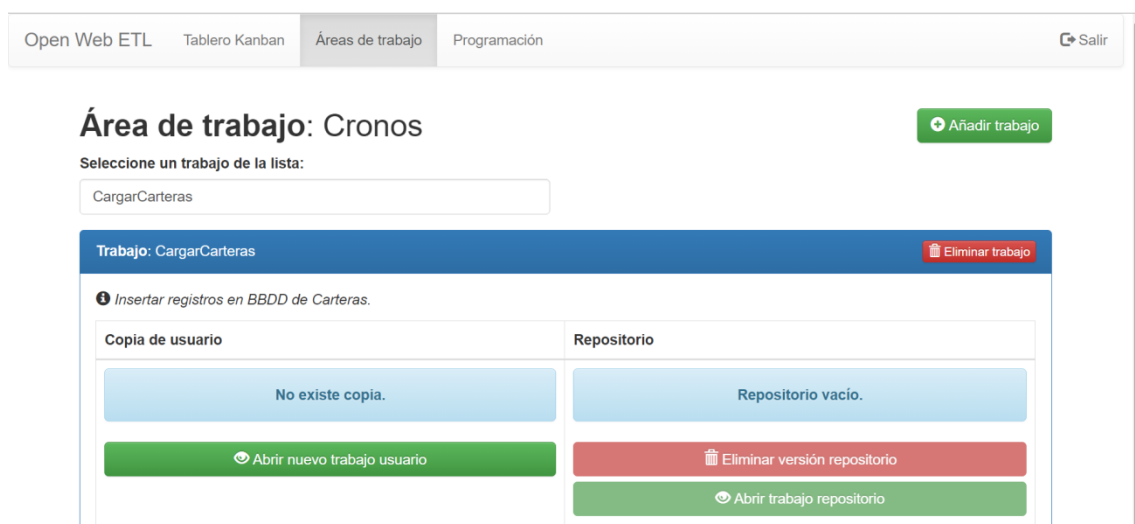


Ilustración 20 - Diseño área de trabajo 2

En el caso de que no exista una copia de usuario del trabajo se mostrará una etiqueta con fondo azul con el texto "No existe copia" en la columna 'Copia de usuario' y justo debajo un botón de color verde con el símbolo de un ojo abierto y de nombre 'Abrir nuevo trabajo usuario'. Del mismo modo, en el caso de que no exista una versión en el repositorio se mostrará una etiqueta con fondo azul con el texto "Repositorio vacío" en la columna 'Repositorio'. Además, los botones 'Eliminar versión repositorio' y 'Abrir trabajo repositorio' aparecerán deshabilitados. Todo ello facilita al usuario la tarea de conocer de forma clara y rápida si existen copias guardadas del trabajo tanto del propio usuario como en el repositorio.

▫ Área para la elaboración de procesos ETL

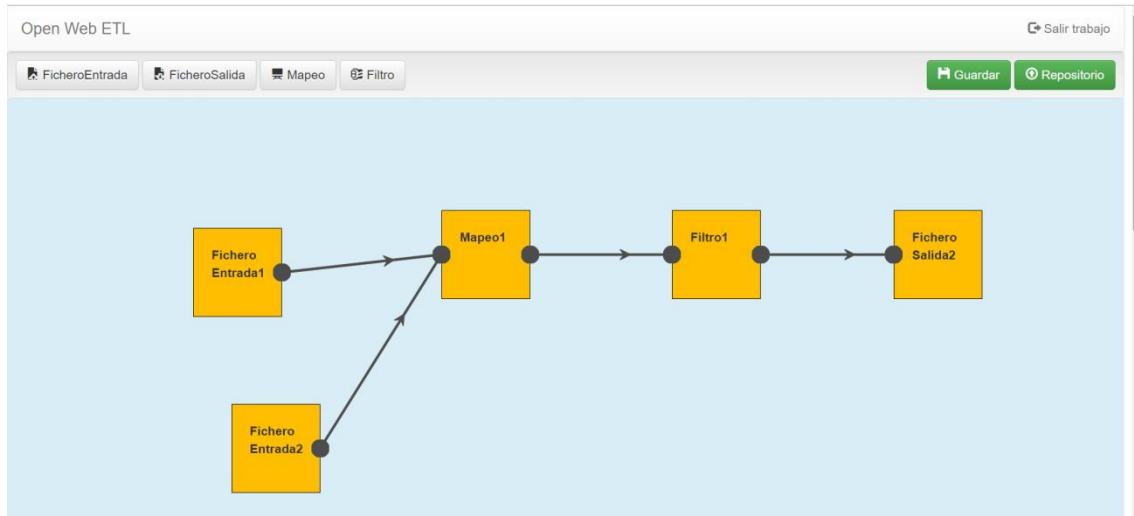


Ilustración 21 - Diseño área de elaboración de proceso ETL

Llegados a este punto, la vista de la página ha cambiado completamente, incluida la barra de navegación común al resto de vistas. En su lugar, la barra de navegación del área para la elaboración de procesos ETL del trabajo al que se ha accedido de la forma en la que se ha descrito antes incluye una barra de navegación con el título de la aplicación alineado a la izquierda y un enlace para regresar a la sección del área de trabajo del trabajo que se está tratando con nombre 'Salir trabajo'. Debajo de esta barra de navegación se sitúa la barra de herramientas para la elaboración de procesos ETL.

Esta barra de herramientas incluye un botón por cada uno de los componentes para la elaboración de procesos ETL que ofrece la herramienta, se encuentran alineados a la izquierda y contienen un símbolo diferente para que el usuario pueda familiarizarse con ellos rápidamente. Debajo de esta barra de herramientas se sitúa el área de trabajo con fondo azul para que el usuario pueda identificarla. Esta área de trabajo contiene la representación visual del proceso ETL en la que cada componente será representado con un contenedor de color amarillo y bordes negros que incluirá el nombre del componente para que el usuario sepa de qué componente se trata. Estos componentes irán conectados por flechas cuyas puntas marcarán la dirección del flujo de datos.

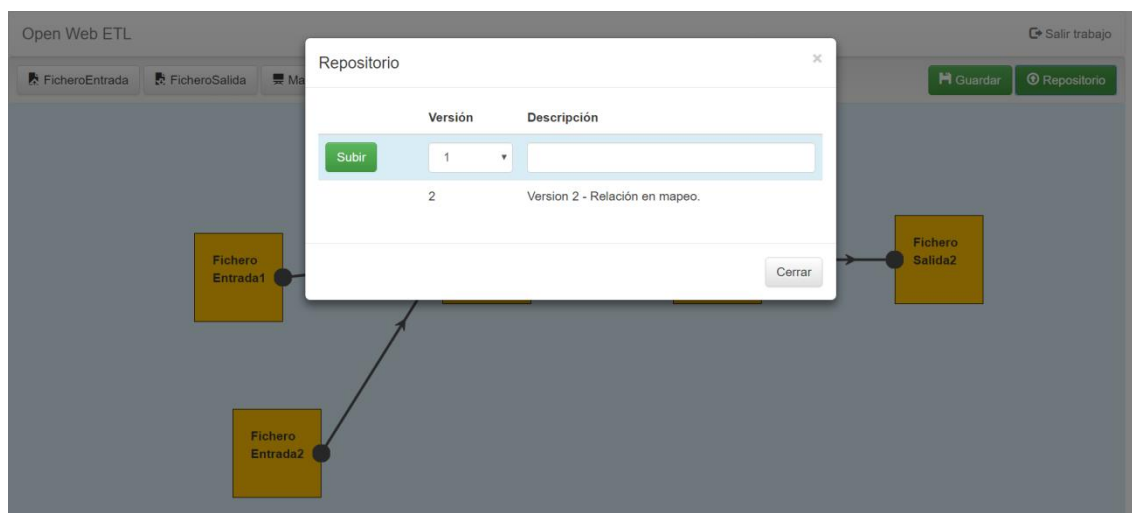


Ilustración 22 - Diseño subida versión

La barra de herramientas descrita contiene además dos botones de color verde alineados a la derecha. El primer botón contiene el símbolo de un disquete y se llama 'Guardar' para que el usuario identifique claramente la acción que ejecuta este botón. Cuando el usuario hace clic en este botón, se guarda el proceso ETL que refleja la vista en ese momento y se muestra una ventana modal informativa de la acción. El segundo botón contiene el símbolo de subida y se llama 'Repositorio' para que el usuario identifique claramente la acción que ejecuta este botón. Cuando el usuario hace clic en este botón, se abre la ventana modal para la subida de versiones al repositorio.

La ventana modal de subida de versiones al repositorio incluye una cabecera con el título 'Repositorio' para situar al usuario en el contexto. El cuerpo de esta ventana incluye una tabla con tres columnas con una cabecera que incluye el nombre 'Versión' en la segunda columna y el nombre 'Descripción' en la tercera. La primera fila está destinada a subir una versión del trabajo al repositorio, para poder distinguirla presenta un fondo de color azul. Esta fila incluye un botón verde con el nombre 'Subir' en la primera columna. Cuando el usuario hace clic en este botón, se sube el trabajo del proceso ETL a la versión del repositorio indicada en el campo selector que se encuentra en la segunda columna. En la tercera columna se encuentra un campo de texto donde el usuario indica la descripción de la subida para de esta forma saber qué cambios incluye. Después de esta acción, los datos de la versión subida se muestran en esta tabla como una fila más. Para salir de esta ventana modal se incluye un botón con nombre 'Cerrar' alineado a la derecha en el pie de la misma.

A continuación, se van a detallar los diseños de cada uno de los componentes que ofrece la herramienta ETL separados por puntos para poder explicarlos de forma más clara. Como ya se ha visto la estructura de diseño seguida durante las vistas ya explicadas, no se va a hacer hincapié en explicar cada uno de los elementos de las vistas de las ventanas modales de configuración de cada uno de los componentes, por el contrario, se darán breves explicaciones acompañadas de ilustraciones con mucho detalle que hablarán por sí mismas.

❖ Componente 'FicheroEntrada'

El primer botón de componentes de la barra de herramientas se llama 'FicheroEntrada'. Haciendo clic en este botón el usuario lo activa y, volviendo a hacer clic en en el área de trabajo, el usuario crea un componente de tipo 'FicheroEntrada'.

El usuario puede abrir la ventana modal de configuración de este componente haciendo doble clic dentro del cuadro amarillo que representa el componente en la vista.

La ventana incluye una cabecera con el título 'Fichero entrada' para poner al usuario en contexto. En el pie de la misma se incluye un botón con nombre 'Cerrar' alineado a la derecha para poder salir de la ventana modal del componente. Por su parte, el cuerpo incluye un botón rojo para poder eliminar el componente y debajo de éste cuatro pestañas que al ser pulsadas cargan debajo una sección de configuración del componente. Estas pestañas son 'Conector', 'Mapeo campos', 'Ver campos' y 'Fichero'. A continuación se muestran los diseños de cada una de estas secciones y debajo una breve descripción:



Ilustración 23 - Diseño conector componente fichero entrada

En la sección 'Conector' el usuario selecciona el componente con el que se va a conectar en la salida y pulsa el botón que se le ofrece para aplicar los cambios. En la vista aparecerá una flecha entre este componente y el que ha elegido el usuario.

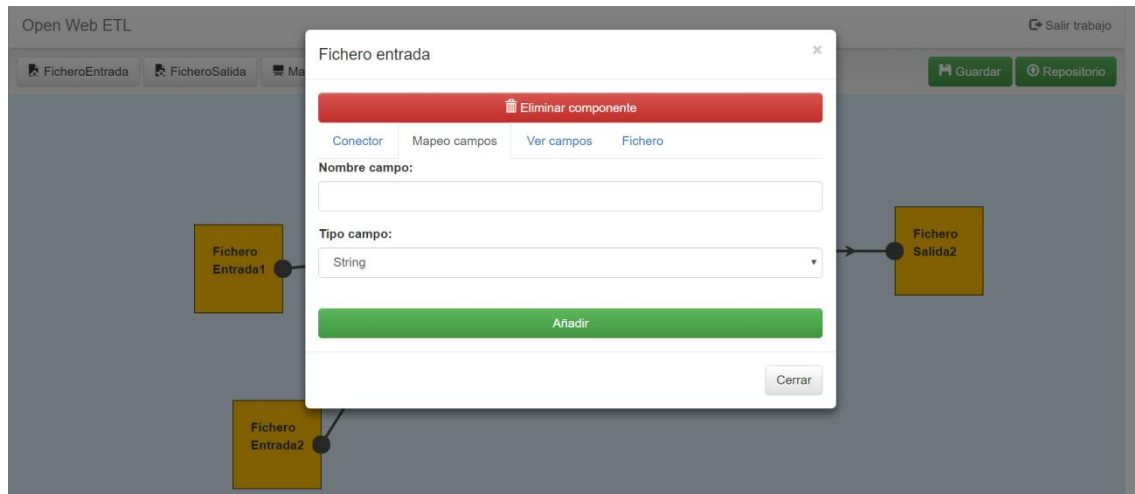


Ilustración 24 - Diseño mapeo campos componente fichero entrada

En la sección 'Mapeo campos' el usuario dispone de un campo de texto para introducir el nombre del campo a mapear. Debajo de éste dispone de un campo selector con los tipos de datos disponibles para que elija el que se corresponda con el campo indicado y, por último, dispone de un botón con nombre 'Añadir' para agregar el nuevo campo.

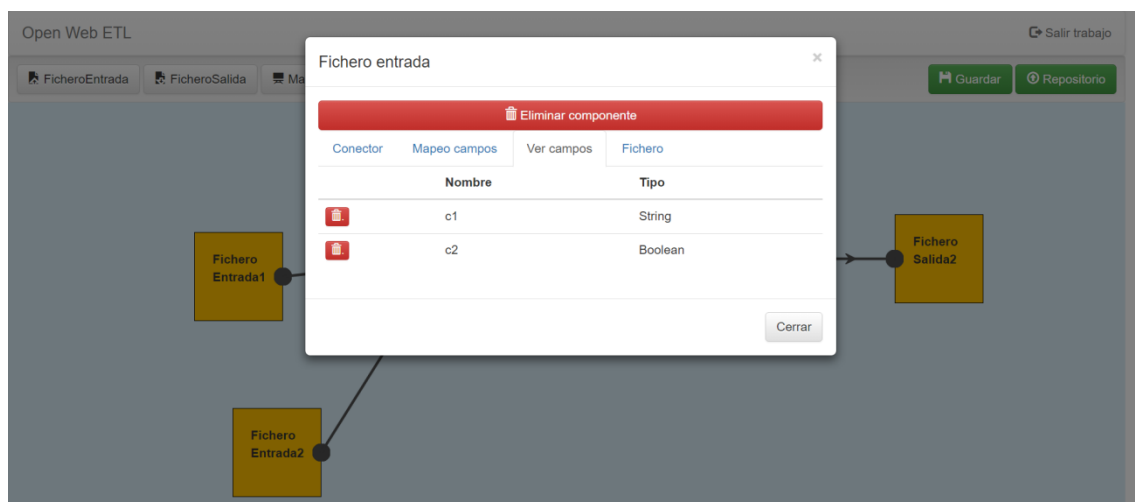


Ilustración 25 - Diseño ver campos componente fichero entrada

En la sección 'Ver campos' el usuario visualiza en una tabla los campos mapeados en el componente. Cada fila representa un campo, donde su segunda columna indica el nombre del campo y la tercera el tipo de dato del mismo. La primera columna incluye un botón que al ser pulsado por el usuario elimina el campo y su fila desaparece de la tabla.

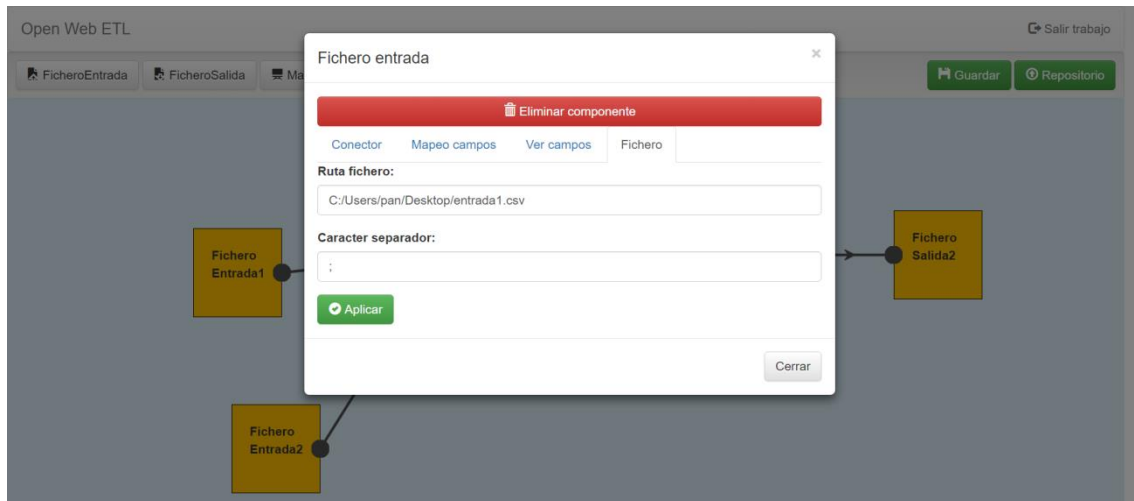


Ilustración 26 - Diseño fichero componente fichero entrada

En la sección 'Fichero' el usuario introduce en un campo de texto la ruta completa donde estará ubicado el fichero de entrada de tipo CSV que representa este componente. Asimismo, introduce en el siguiente campo de texto el carácter que separará cada columna de dicho fichero y hace clic en el botón 'Aplicar' para confirmar la operación.

❖ Componente 'FicheroSalida'

El segundo botón de componentes de la barra de herramientas se llama 'FicheroSalida'. Haciendo clic en este botón el usuario lo activa y, volviendo a hacer clic en en el área de trabajo, el usuario crea un componente de tipo 'FicheroSalida'.

El usuario puede abrir la ventana modal de configuración de este componente haciendo doble clic dentro del cuadro amarillo que representa el componente en la vista.

La ventana incluye una cabecera con el título 'Fichero salida' para poner al usuario en contexto. En el pie de la misma se incluye un botón con nombre 'Cerrar' alineado a la derecha para poder salir de la ventana modal del componente. Por su parte, el cuerpo incluye un botón rojo para poder eliminar el componente y debajo de éste dos pestañas que al ser pulsadas cargan debajo una sección de configuración del componente. Estas pestañas son 'Ver campos' y 'Fichero'. A continuación se muestran los diseños de cada una de estas secciones y debajo una breve descripción:

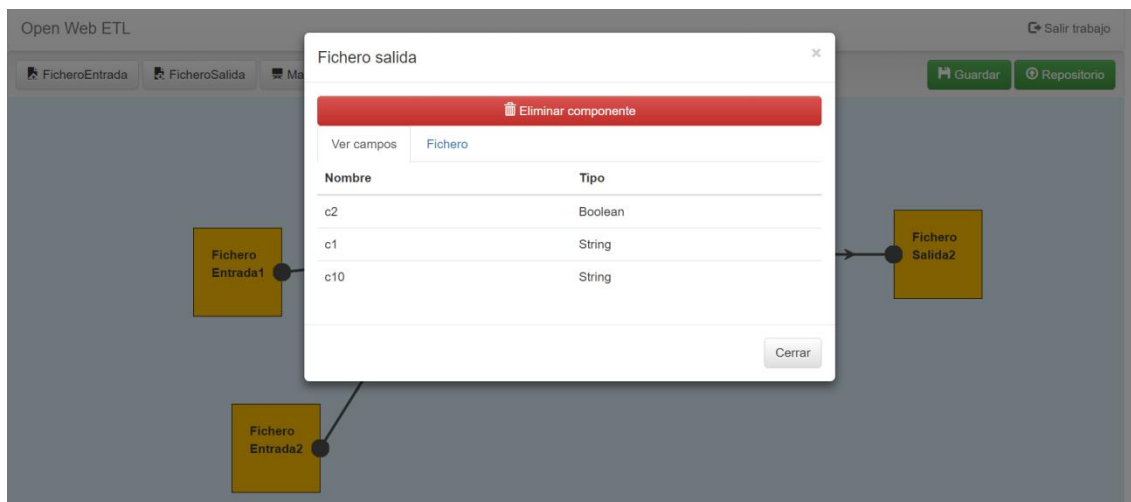


Ilustración 27 - Diseño ver campos fichero salida

En la sección 'Ver campos' el usuario visualiza en una tabla los campos mapeados en el componente. Cada fila representa un campo, donde su primera columna indica el nombre del campo y la segunda el tipo de dato del mismo.

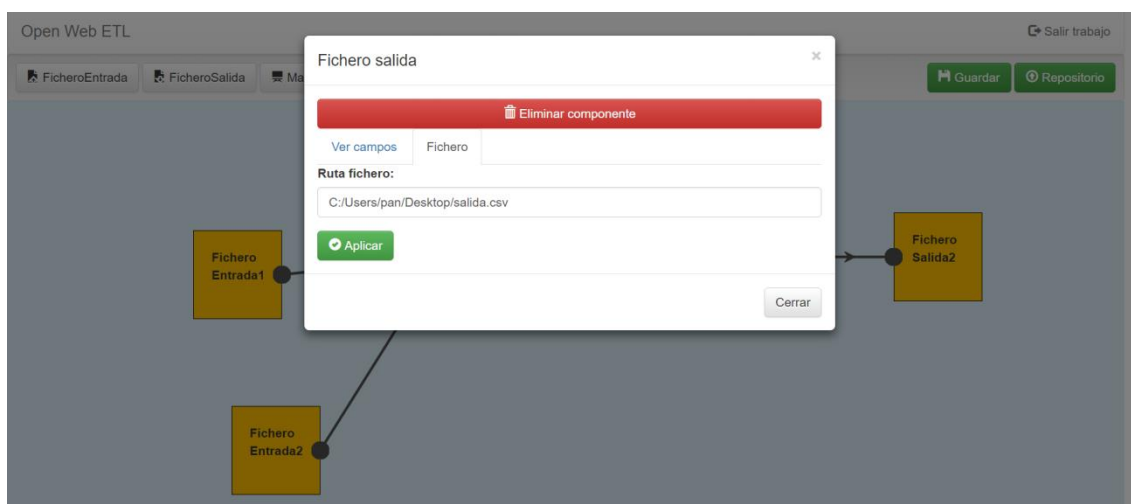


Ilustración 28 - Diseño fichero componente fichero salida

En la sección 'Fichero' el usuario introduce en un campo de texto la ruta completa donde estará ubicado el fichero de salida de tipo CSV que representa este componente y hace clic en el botón 'Aplicar' para confirmar la operación.

❖ Componente 'Mapeo'

El tercer botón de componentes de la barra de herramientas se llama 'Mapeo'. Haciendo clic en este botón el usuario lo activa y, volviendo a hacer clic en el área de trabajo, el usuario crea un componente de tipo 'Mapeo'.

El usuario puede abrir la ventana modal de configuración de este componente haciendo doble clic dentro del cuadro amarillo que representa el componente en la vista.

La ventana incluye una cabecera con el título 'Mapeo' para poner al usuario en contexto. En el pie de la misma se incluye un botón con nombre 'Cerrar' alineado a la derecha para poder salir de la ventana modal del componente. Por su parte, el cuerpo incluye un botón rojo para poder eliminar el componente y debajo de éste tres pestañas que al ser pulsadas cargan debajo una sección de configuración del componente. Estas pestañas son 'Conector', 'Mapeo campos' y 'Relaciones'. A continuación se muestran los diseños de cada una de estas secciones y debajo una breve descripción:

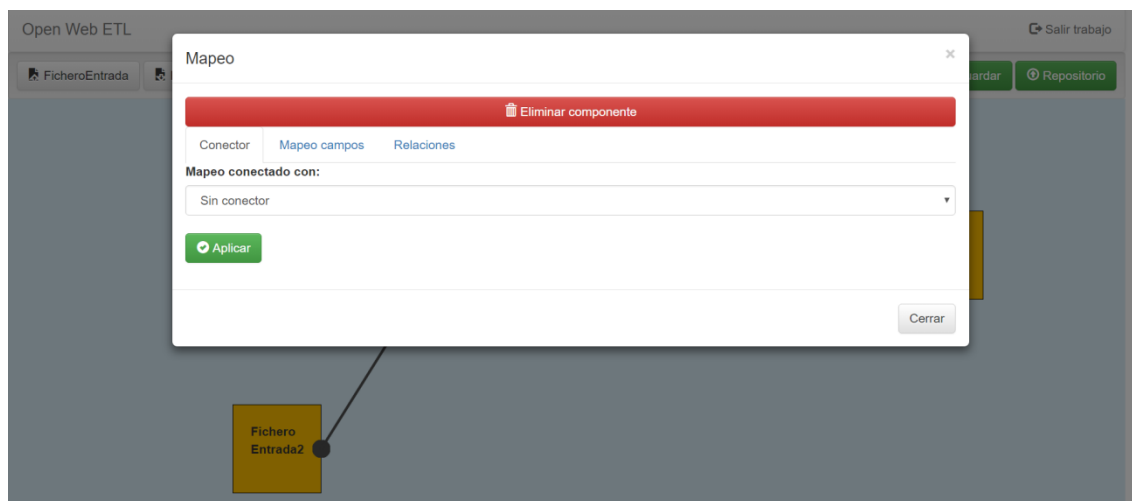


Ilustración 29 - Diseño conector componente mapeo

En la sección 'Conector' el usuario selecciona el componente con el que se va a conectar en la salida y pulsa el botón que se le ofrece para aplicar los cambios. En la vista aparecerá una flecha entre este componente y el que ha elegido el usuario.

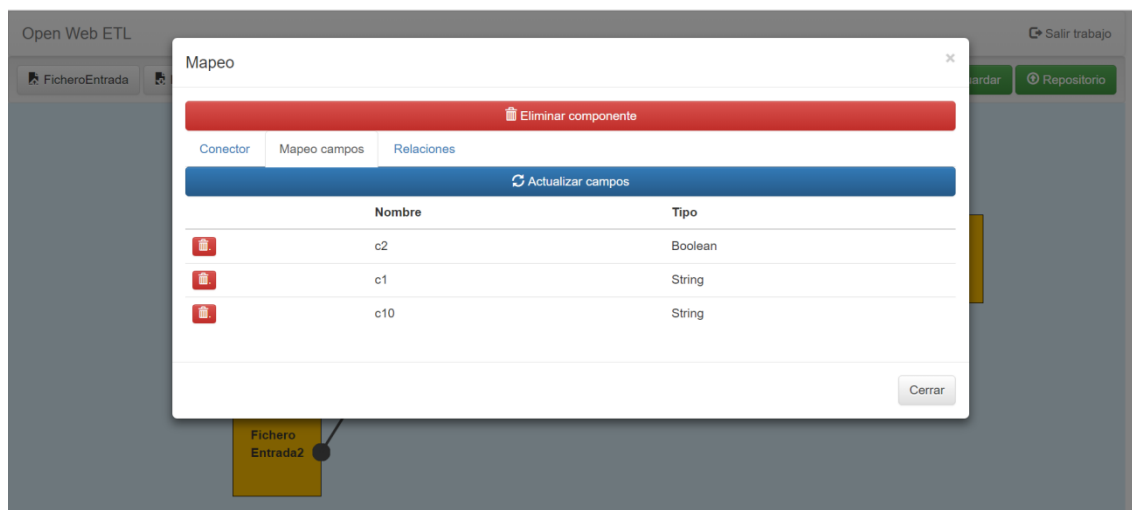


Ilustración 30 - Diseño mapeo campos componente mapeo

En la sección 'Mapeo campos' el usuario visualiza en una tabla los campos mapeados en el componente. Cada fila representa un campo, donde su segunda columna indica el nombre del campo y la tercera el tipo de dato del mismo. La primera columna incluye un botón que al ser

pulsado por el usuario elimina el campo y su fila desaparece de la tabla. Encima de esta tabla de campos se encuentra un botón 'Actualizar campos' que actualiza la tabla de campos con los que recibe de los componentes con los que está conectado el presente a su entrada.

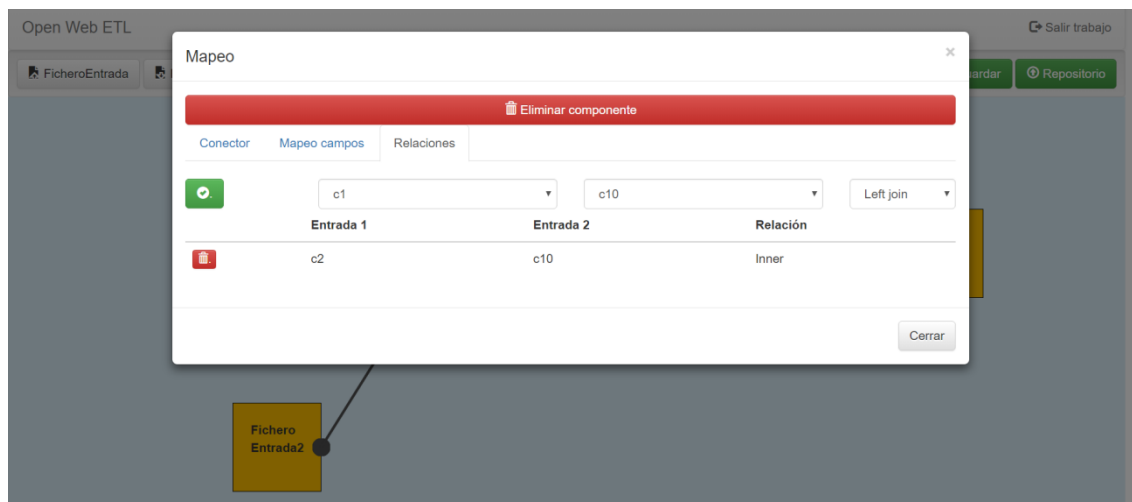


Ilustración 31 - Diseño relaciones componente mapeo

En la sección 'Relaciones', en el caso de que existan dos ficheros de entrada conectados a este componente, el usuario visualiza las relaciones existentes entre los campos del primer fichero y el segundo en una tabla, donde cada fila es una relación. Cada fila incluye un botón en la primera columna para borrar la relación y hacer que la fila desaparezca de esta tabla. Encima de esta tabla el usuario dispone de un campo selector para elegir un campo del primer fichero, un segundo campo selector para elegir un campo del segundo fichero y un tercer campo selector para elegir el tipo de relación ('Left join', 'Inner join') entre ambos campos. A estos campos selectores les precede un botón para añadir la relación indicada y hacer que se muestre como una fila más en la tabla de relaciones mencionada.

❖ Componente 'Filtro'

El cuarto botón de componentes de la barra de herramientas se llama 'Filtro'. Haciendo clic en este botón el usuario lo activa y, volviendo a hacer clic en en el área de trabajo, el usuario crea un componente de tipo 'Filtro fila'.

El usuario puede abrir la ventana modal de configuración de este componente haciendo doble clic dentro del cuadro amarillo que representa el componente en la vista.

La ventana incluye una cabecera con el título 'Filtro fila' para poner al usuario en contexto. En el pie de la misma se incluye un botón con nombre 'Cerrar' alineado a la derecha para poder salir de la ventana modal del componente. Por su parte, el cuerpo incluye un botón rojo para poder eliminar el componente y debajo de éste tres pestañas que al ser pulsadas cargan debajo una sección de configuración del componente. Estas pestañas son 'Conector', 'Mapeo campos' y 'Filtros'. A continuación se muestran los diseños de cada una de estas secciones y debajo una breve descripción:

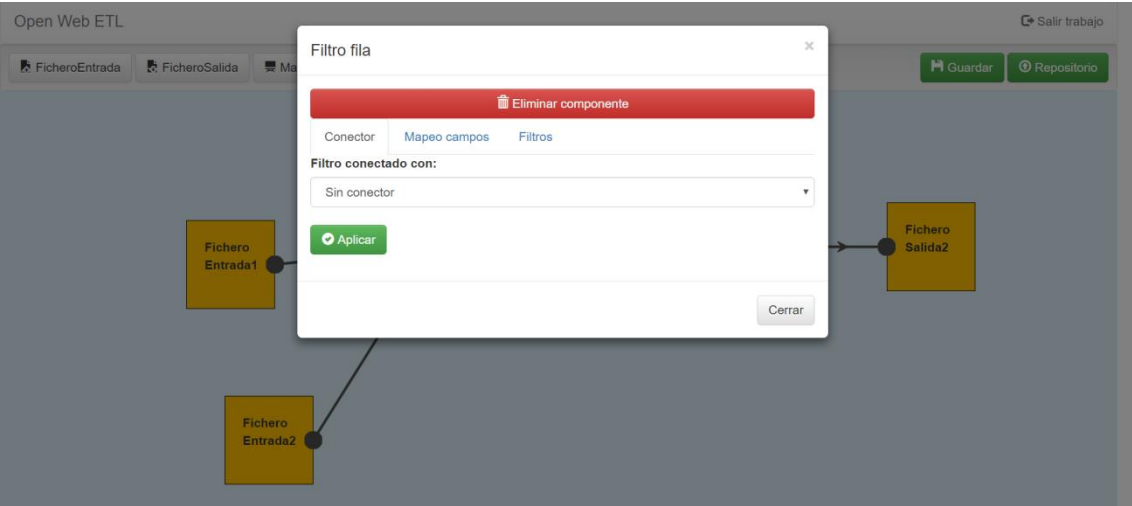


Ilustración 32 - Diseño conector componente filtro

En la sección 'Conector' el usuario selecciona el componente con el que se va a conectar en la salida y pulsa el botón que se le ofrece para aplicar los cambios. En la vista aparecerá una flecha entre este componente y el que ha elegido el usuario.

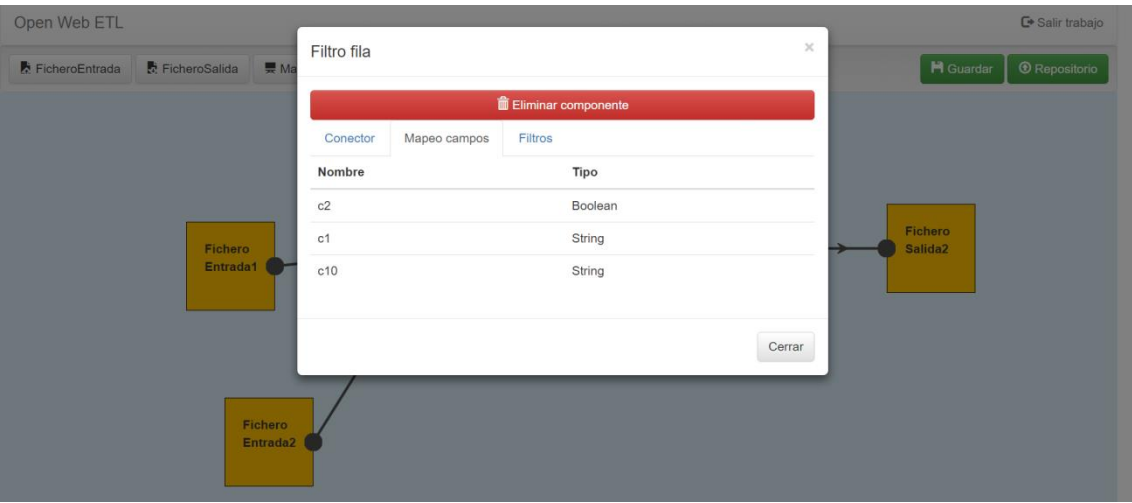


Ilustración 33 - Diseño mapeo campos componente filtro

En la sección 'Mapeo campos' el usuario visualiza en una tabla los campos mapeados en el componente. Cada fila representa un campo, donde su primera columna indica el nombre del campo y la segunda el tipo de dato del mismo.

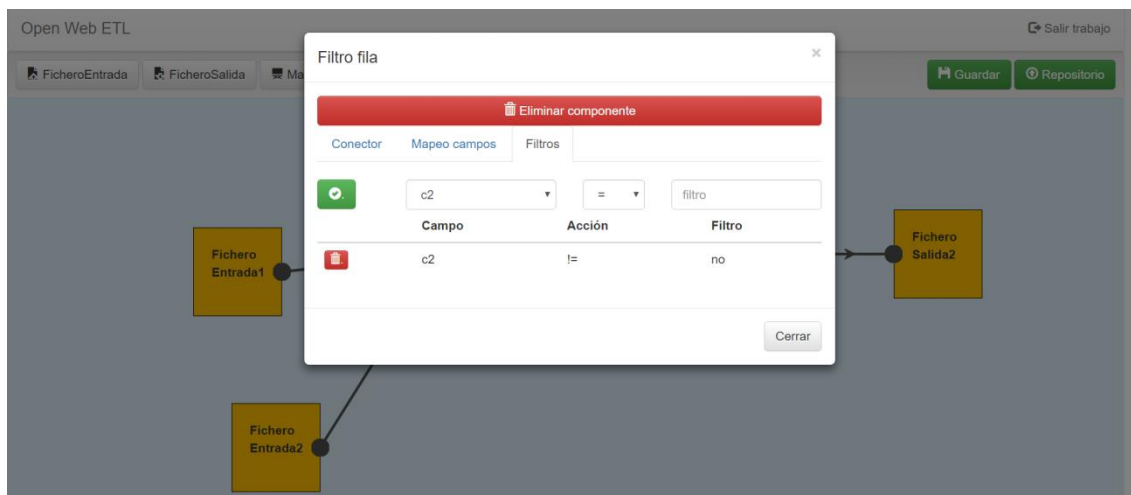


Ilustración 34 - Diseño filtros componente filtro

En la sección 'Filtros' el usuario visualiza los filtros de filas existentes en una tabla, cada fila representa un filtro. Cada una de estas filas incluye un botón en la primera columna para borrar el filtro y hacer que la fila desaparezca de esta tabla. Encima de esta tabla el usuario dispone de un campo selector para elegir un campo mapeado, un segundo campo selector para elegir el tipo de filtro (distinto o igual) y un tercer campo de texto para indicar el valor con el que comparar el campo indicado. A estos campos les precede un botón para añadir el filtro indicado y hacer que se muestre como una fila más en la tabla de filtros mencionada.

○ Programación



Ilustración 35 - Diseño programación trabajos

La sección para programar y ejecutar los trabajos se encuentra justo debajo de la barra de navegación y comienza con el título 'Programación de procesos' que ayuda al usuario a saber en qué punto de la aplicación se encuentra. Debajo de este título se sitúa una tabla con las programaciones. Esta tabla consta de las siguientes columnas de izquierda a derecha: una botonera de acciones, nombre del trabajo, versión del trabajo, horario de programación del trabajo, fecha de la última ejecución del trabajo, estado de la última ejecución y, en su última columna, descripción de la última ejecución. Contiene una cabecera que describe cada una de estas columnas.

La primera fila de la tabla sirve para añadir una programación a un trabajo existente, para diferenciarla del resto su fondo es de color azul. La primera columna de esta fila incluye un botón

verde con un símbolo de confirmación para que el usuario identifique rápidamente la acción que ejecuta este botón. Cuando el usuario hace clic en este botón, se guarda la programación del trabajo y se muestra una ventana modal informativa. Acto seguido, se añade una fila con esta nueva programación al final de la tabla de programaciones. La segunda columna de esta fila incluye un campo selector donde aparecen listados los nombres de los trabajos de las áreas de trabajo a las que está vinculado el usuario de la aplicación. La tercera columna incluye un campo selector con un listado de número del uno al cinco para que el usuario elija el número de versión a programar. Por último, la cuarta columna incluye un campo de texto donde el usuario indica el horario de programación siguiendo el formato soportado por la librería Quartz Scheduler. El resto de columnas no incluyen campos.

El resto de filas de la tabla de programación muestran las programaciones y la información de las últimas ejecuciones de sus trabajos. Incluyen un botón de color azul con el símbolo *play* para que el usuario identifique rápidamente la acción que ejecuta este botón. Cuando el usuario hace clic en este botón, se ejecuta el proceso ETL asociado a la versión del trabajo de su fila y se muestra una ventana modal informativa. Acto seguido, se actualizan las celdas de la fila correspondientes a las columnas 'Última ejecución', 'Estado' y 'Descripción', para que de esta manera el usuario conozca de forma casi instantánea el resultado de la ejecución del trabajo de esta fila y pueda obrar en consecuencia. Junto al botón anterior, se sitúa un botón de color rojo con el símbolo de una papelera para que el usuario identifique rápidamente la acción que ejecuta este botón. Cuando el usuario hace clic en este botón, se abre una ventana modal de confirmación para eliminar la programación indicada en la fila donde está situado este botón. Acto seguido, desaparecerá de la tabla de programaciones la fila mencionada.

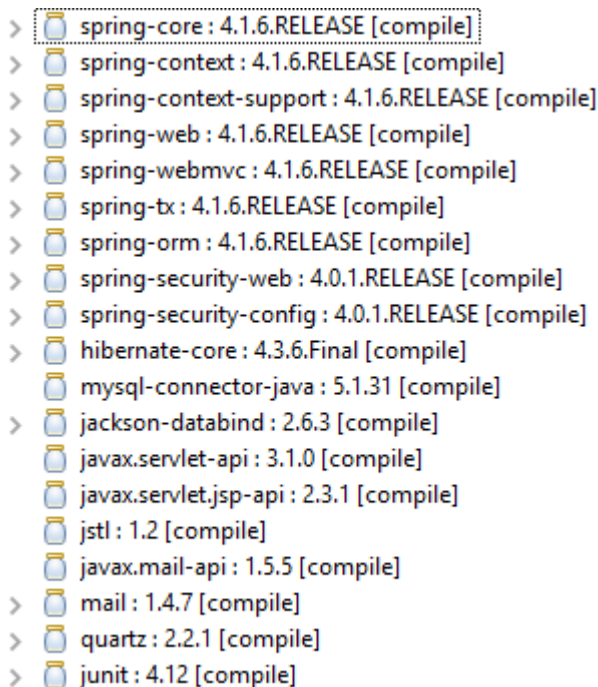
4.2.3. Implementación

Debido al gran volumen de funcionalidades a implementar y a la complejidad de esa implementación, se ha acordado con el cliente que este apartado incluya la información necesaria para poder implementar la aplicación, pero asumiendo que puede que sea necesario en un futuro redactar documentos que detallen con más precisión la implementación de cada una de las capas de negocio que conforman la herramienta.

La implementación consiste en la construcción de una aplicación Spring MVC protegida mediante Spring Security, donde la integración con la base de datos MySQL se lleva a cabo empleando Hibernate. El almacenamiento de contraseñas se realiza en formato cifrado utilizando BCrypt.

Se empleará Eclipse como entorno de desarrollo integrado, concretamente Eclipse Java EE IDE (<<Mars>>) para poder crear la aplicación web en Java, este último es el lenguaje de programación que debe ser utilizado para realizar la implementación. En este caso será necesario tener instalado la versión JDK 7. Desde este IDE se debe crear un proyecto Maven con nombre 'openwebetl'.

Mediante Maven se gestionarán las librerías necesarias para la implementación, así como la versión exacta que se necesite, sin preocuparse de si una librería depende de otras para poder funcionar de manera óptima. Se requerirá definir un Artefacto que contendrá toda la información requerida para una correcta gestión. Este Artefacto se creará en un fichero pom.xml que tendrá que incluir las librerías que se muestran en la siguiente ilustración y sus dependencias:






















- >  spring-core : 4.1.6.RELEASE [compile]
- >  spring-context : 4.1.6.RELEASE [compile]
- >  spring-context-support : 4.1.6.RELEASE [compile]
- >  spring-web : 4.1.6.RELEASE [compile]
- >  spring-webmvc : 4.1.6.RELEASE [compile]
- >  spring-tx : 4.1.6.RELEASE [compile]
- >  spring-orm : 4.1.6.RELEASE [compile]
- >  spring-security-web : 4.0.1.RELEASE [compile]
- >  spring-security-config : 4.0.1.RELEASE [compile]
- >  hibernate-core : 4.3.6.Final [compile]
-  mysql-connector-java : 5.1.31 [compile]
- >  jackson-databind : 2.6.3 [compile]
-  javax.servlet-api : 3.1.0 [compile]
-  javax.servlet.jsp-api : 2.3.1 [compile]
-  jstl : 1.2 [compile]
-  javax.mail-api : 1.5.5 [compile]
- >  mail : 1.4.7 [compile]
- >  quartz : 2.2.1 [compile]
- >  junit : 4.12 [compile]

Ilustración 36 - Librería requeridas para la implementación.

La estructura final del proyecto será la siguiente:

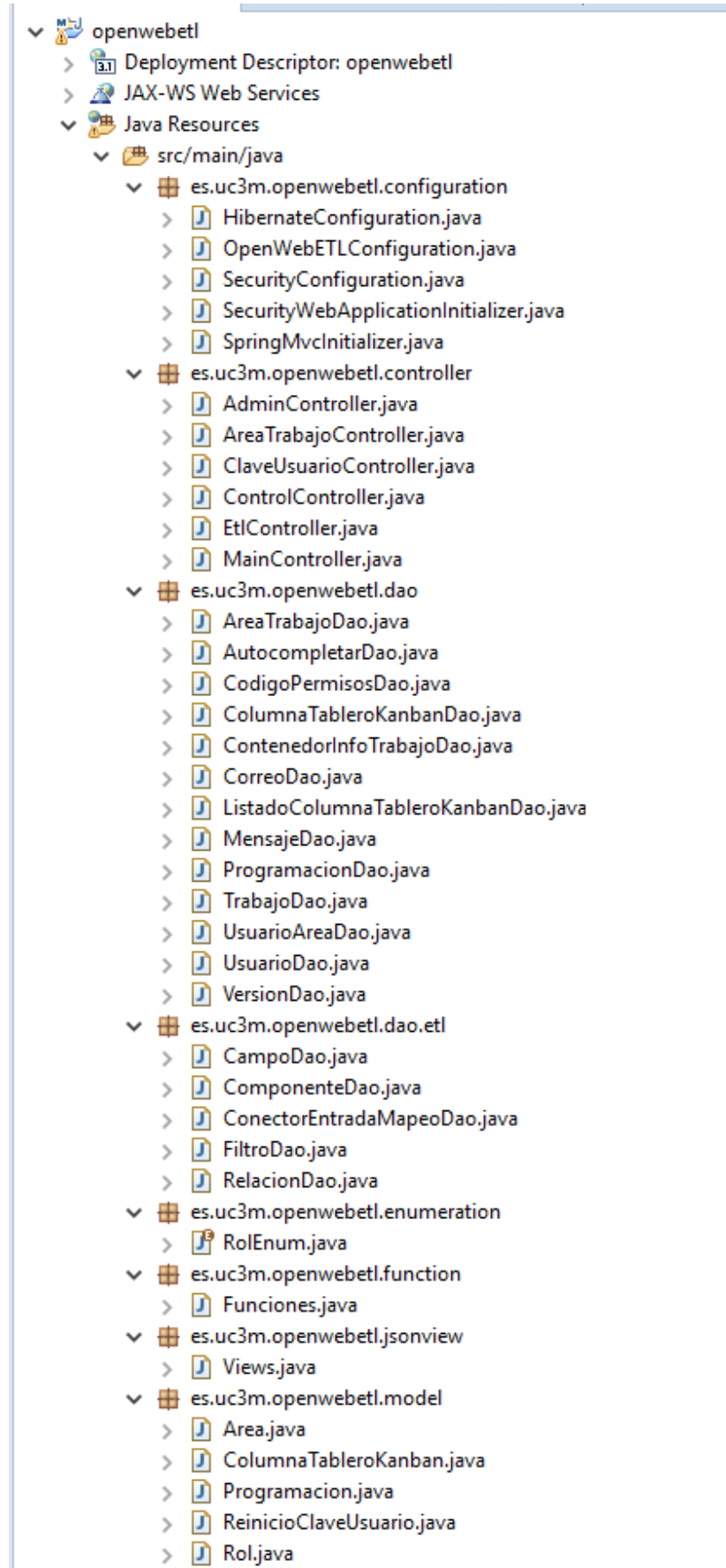


Ilustración 37- Estructura de ficheros en Eclipse, parte 1

- > TipoColumnaTableroKanban.java
- > Trabajo.java
- > Usuario.java
- > VersionTrabajoRepositorio.java
- > VersionTrabajoUsuario.java
- ▼ es.uc3m.openwebetl.model.etl
 - > CampoFicheroEntrada.java
 - > CampoFicheroSalida.java
 - > CampoFiltroFila.java
 - > CampoMapeo.java
 - > ConectorEntradaMapeo.java
 - > FicheroEntrada.java
 - > FicheroSalida.java
 - > FiltroFila.java
 - > FiltroFiltroFila.java
 - > Mapeo.java
 - > RelacionMapeo.java
- ▼ es.uc3m.openwebetl.service
 - > AccesoService.java
 - > AdminService.java
 - > AreaTrabajoService.java
 - > ClaveUsuarioService.java
 - > CustomUserDetailsService.java
 - > EtlService.java
 - > MailService.java
- ▼ src/main/resources
 - application.properties
- > src/test/java
- > src/test/resources
- > Libraries
- > JavaScript Resources
- > Deployed Resources

Ilustración 38 - Estructura de ficheros en Eclipse, parte 2

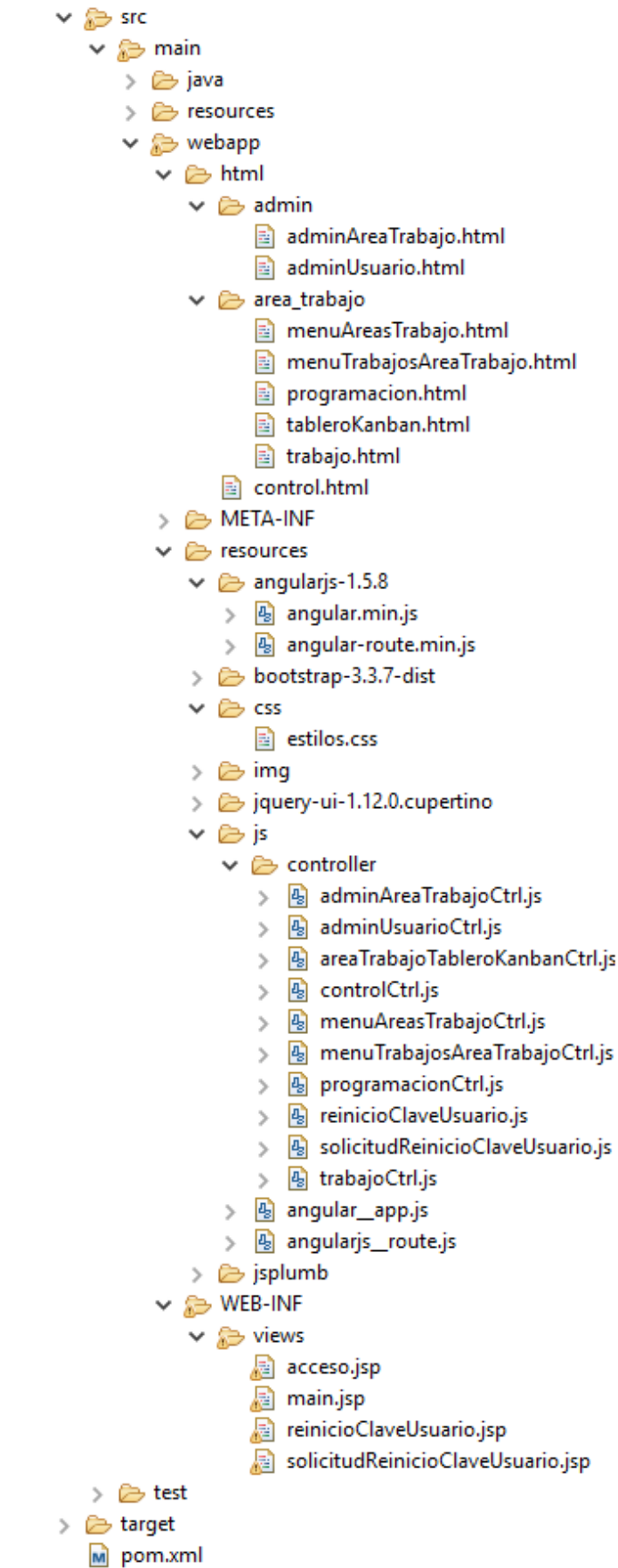


Ilustración 39 - Estructura de ficheros en Eclipse, parte 3

A continuación, se detalla el contenido de la estructura de directorios del proyecto mostrada en la ilustración anterior. Para ello, se presentan una serie de apartados donde cada uno detallará una capa de negocio o configuración de Spring MVC, Spring Security y Hibernate.

4.2.3.1. Configuración de Spring MVC, Spring Security y Hibernate

Será necesario implementar las clases que permitan configurar los frameworks Spring MVC, Spring Security y Hibernate. Con este fin, se creará un paquete con nombre 'es.uc3m.openwebetl.configuration' que contendrá las clases que se encargarán de esta configuración y que se indican a continuación. Se resaltarán anotaciones y métodos de cada clase, pero se asume que las personas que se hagan cargo de la implementación tienen conocimientos suficientes acerca de estos frameworks.

- **HibernateConfiguration.java**

En esta clase se configurará Hibernate mediante el empleo de las siguientes anotaciones:

`@Configuracion` indica que esta clase contiene uno o más métodos bean anotados con `@Bean`, generando beans manejables por el contenedor de Spring. En este caso, esta clase representa la configuración de Hibernate.

`@ComponentScan` indica el paquete en el que se encuentran los controladores (`@Controller`).

`@EnableTransactionManagement` habilita el manejo de transacciones.

`@PropertySource` es empleada para declarar un conjunto de propiedades (definidos en un archivo de propiedades) en `Environment`, proporcionando flexibilidad para tener distintos valores en diferentes entornos de aplicación. Es por este motivo por el que no es necesario que la aplicación y la base de datos estén en el mismo servidor. Es suficiente con que en este archivo de propiedades se aporte la información necesaria para que la aplicación pueda trabajar con la base de datos.

El método `sessionFactory()` crea un `LocalSessionFactoryBean`, para lo que se necesita indicar la fuente de datos y las propiedades de Hibernate. Mediante la anotación `@PropertySource` se puede externalizar los valores reales en un archivo de propiedades y usar `Environment` de Spring para recuperar el valor correspondiente a un elemento. Una vez creada la `SessionFactory`, ésta se inyectará en el método Bean `transactionManager` que puede llegar a dar soporte a las transacciones para las sesiones creadas por este `sessionFactory`.

A continuación, se muestra el contenido que debe tener el fichero de propiedades mencionado anteriormente con nombre 'application.properties' y ubicado en '/src/main/resources':

```
jdbc.driverClassName = com.mysql.jdbc.Driver
jdbc.url = jdbc:mysql://[valor que corresponda]/openwebetl
jdbc.username = [valor que corresponda]
jdbc.password = [valor que corresponda]
hibernate.dialect = org.hibernate.dialect.MySQLDialect
hibernate.show_sql = false
hibernate.format_sql = false

mail.host = smtp.gmail.com
mail.port = 587
mail.username = [valor que corresponda]
```

```
mail.password = [valor que corresponda]
mail.smtp.from = [valor que corresponda]
mail.smtp.starttls.enable = true
mail.smtp.auth = true
mail.transport.protocol = smtp
mail.debug = false

aplicacion.url = http:// [valor que corresponda]/openwebetl
```

- **OpenWebETLConfiguration.java**

@Configuración indica que esta clase contiene uno o más métodos bean anotados con @Bean, generando beans manejables por el contenedor de Spring.

Mediante la anotación @ComponentScan se hace referencia a la localización del paquete en el que se encuentran los beans asociados.

La anotación @EnableWebMvc es empleada para habilitar Spring MVC.

El método `public ViewResolver viewResolver()` configura una ViewResolver para identificar dónde se encuentran realmente las vistas.

El método `void addResourceHandlers(ResourceHandlerRegistry registry)` configura ResourceHandlers para servir recursos estáticos como CSS, JavaScript, etcétera.

El método `public JavaMailSender getMailSender()` realiza la configuración para poder enviar correos electrónicos. Los datos de configuración los obtiene del fichero de configuración 'application.properties' detallado anteriormente.

- **SecurityConfiguration.java**

En esta clase se configura la seguridad de la aplicación estableciendo BCrypt como base para cifrar las contraseñas almacenadas. Asimismo, se recuperan los datos de usuario de base de datos a través de un servicio para determinar la validez y el tipo de rol según los datos de acceso proporcionados. Por último, se restringe el acceso a las URLs de la siguiente manera:

"/": puede acceder todo el mundo.

"/clave/**": puede acceder todo el mundo.

"/main/**": Administrador o Usuario.

"/admin/**": Administrador.

"/area_trabajo/**": Usuario.

"/login": puede acceder todo el mundo. Es la página de acceso. En el caso de acceder a una página restringida, el usuario será redirigida a ésta. Además, la desconexión de la aplicación implicará regresar a esta página.

Del mismo modo, se indican los campos de usuario ("nick") y contraseña ("clave") empleados para poder validar el acceso a las áreas restringidas de la aplicación.

- **SecurityWebApplicationInitializer.java**

Esta clase se emplea para inicializar el módulo de Spring Security dentro de la aplicación.

- **SpringMvcInitializer.java**

En esta clase se registra la *Configuration Class* pudiendo prescindir del fichero *web.xml* necesario en versiones anteriores de Spring.

A continuación se ilustran las clases de configuración con el detalle de sus campos y métodos:

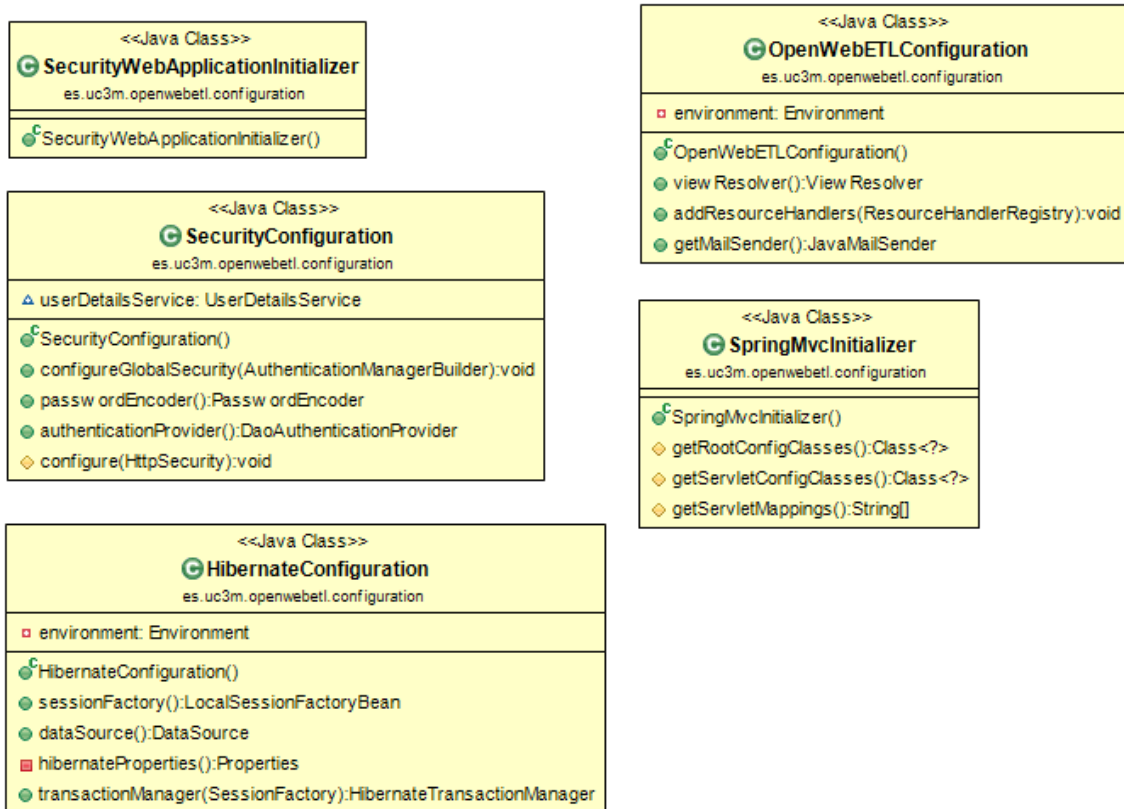


Ilustración 40 - Clases de configuración

4.2.3.2. Capa de negocio: Modelo

El Modelo representa la parte de la aplicación que implementa la lógica de negocio. Es decir, es responsable de la recuperación de datos convirtiéndolos en conceptos significativos para la aplicación, así como su procesamiento, validación, asociación y cualquier otra tarea relativa a la manipulación de dichos datos [84].

Esta capa de negocio estará representada por las clases ubicadas dentro de los paquetes 'es.uc3m.openwebetl.model' y 'es.uc3m.openwebetl.model.etl'. Las clases que contienen estos paquetes se podrán persistir en la base de datos de la aplicación, con este fin, cada clase deberá indicar la tabla donde se persistirá. Aquellas correspondientes al segundo paquete mencionado contendrán clases directamente relacionadas con la funcionalidad de elaboración de procesos ETL.

Es necesario que el equipo de trabajo que implemente la aplicación tenga conocimientos sobre Hibernate y el uso de anotaciones. Con esta premisa, antes de presentar las clases que conforman el Modelo, se van a dar unas pautas de cómo se deberán implementar este tipo de clases. Siguiendo estas pautas junto con la descripción de cada clase y el listado de los campos

y métodos de cada una de ellas, se podrán implementar las clases de la capa de negocio del Modelo.

Estas clases deberán incluir las siguientes anotaciones que se detallan a continuación:

`@Entity` indica que la clase en la que se encuentre es una entidad que se puede persistir.

`@Table(name="nombreDeLaTablaEnBBDD")` indica el nombre de la tabla de la base de datos donde se persistirá la clase.

`@Id` indica que el atributo al que está asociado es la clave primaria.

`@Column(name="nombreCampoEnBBDD")` indica que la propiedad Java que aparece asociada a esta anotación se persistirá en una columna llamada "nombreCampoEnBBDD".

Es importante señalar que se deberá implementar el esquema de base de datos de la aplicación siguiendo el modelo Entidad/Relación mostrado en el apartado dedicado al diseño. Mediante este modelo se deberán identificar las relaciones que existen entre las distintas tablas. Para poder representar estas relaciones en las clases del Modelo se emplearán las siguientes anotaciones:

`@ManyToMany` permite indicar una relación de muchos a muchos entre tablas.

`@OneToOne` permite indicar una relación uno a uno entre tablas.

Una vez dadas las pautas de cómo se deberán implementar este tipo de clases, se detalla cada una de las clases implicadas en esta capa de negocio:

- **Area.java:** esta clase se persistirá en la tabla 'AREATRABAJO'.
- **ColumnaTableroKanban.java:** esta clase se persistirá en la tabla 'COLUMNATABLEROKANBAN'.
- **Programacion.java:** esta clase se persistirá en la tabla 'PROGRAMACION'.
- **ReinicioClaveUsuario.java:** esta clase se persistirá en la tabla 'REINICIO_CLAVE_USUARIO'.
- **Rol.java:** esta clase se persistirá en la tabla 'ROL'.
- **TipoColumnaTableroKanban.java:** esta clase se persistirá en la tabla 'TIPOCOLUMNATABLEROKANBAN'.
- **Trabajo.java:** esta clase se persistirá en la tabla 'TRABAJO'.
- **Usuario.java:** esta clase se persistirá en la tabla 'USUARIO'.
- **VersionTrabajoRepositorio.java:** esta clase se persistirá en la tabla 'VERSION_TRABAJO_REPOSITORIO'.
- **VersionTrabajoUsuario.java:** esta clase se persistirá en la tabla 'VERSION_TRABAJO_USUARIO'.

A continuación se ilustran las clases del Modelo con el detalle de sus campos y métodos:

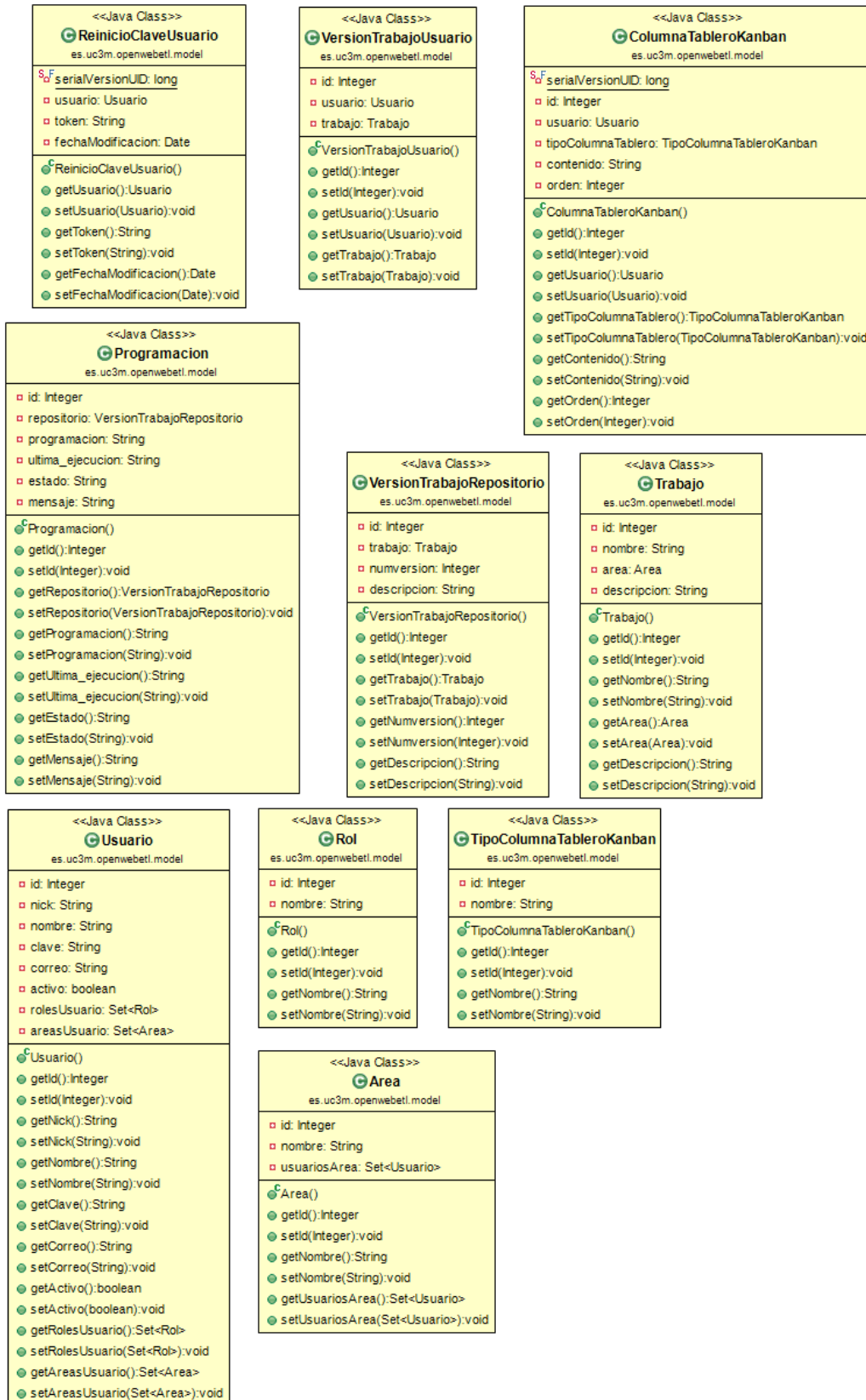


Ilustración 41 - clases del Modelo

Las siguientes clases están asociadas a la funcionalidad de la elaboración de procesos ETL:

- **CampoFicheroEntrada.java:** esta clase se persistirá en la tabla 'CAMPO_FICHEROENTRADA'.
- **CampoFicheroSalida.java:** esta clase se persistirá en la tabla 'CAMPO_FICHEROSALIDA'.
- **CampoFiltroFila.java:** esta clase se persistirá en la tabla 'CAMPO_FILTROFILA'.
- **CampoMapeo.java:** esta clase se persistirá en la tabla 'CAMPO_MAPEO'.
- **ConectorEntradaMapeo.java:** esta clase se persistirá en la tabla 'CONECTORENTRADA_MAPEO'.
- **FicheroEntrada.java:** esta clase se persistirá en la tabla 'COMPONENTE_FICHEROENTRADA'.
- **FicheroSalida.java:** esta clase se persistirá en la tabla 'COMPONENTE_FICHEROSALIDA'.
- **FiltroFila.java:** esta clase se persistirá en la tabla 'COMPONENTE_FILTROFILA'.
- **FiltroFiltroFila.java:** esta clase se persistirá en la tabla 'FILTRO_FILTROFILA'.
- **Mapeo.java:** esta clase se persistirá en la tabla 'COMPONENTE_MAPEO'.
- **RelacionMapeo.java:** esta clase se persistirá en la tabla 'RELACION_MAPEO'.

A continuación se ilustran las clases del Modelo centradas en el marco de la elaboración de procesos ETL con el detalle de sus campos y métodos. Aplicar zoom en la siguiente ilustración para poder visualizar la información correctamente:

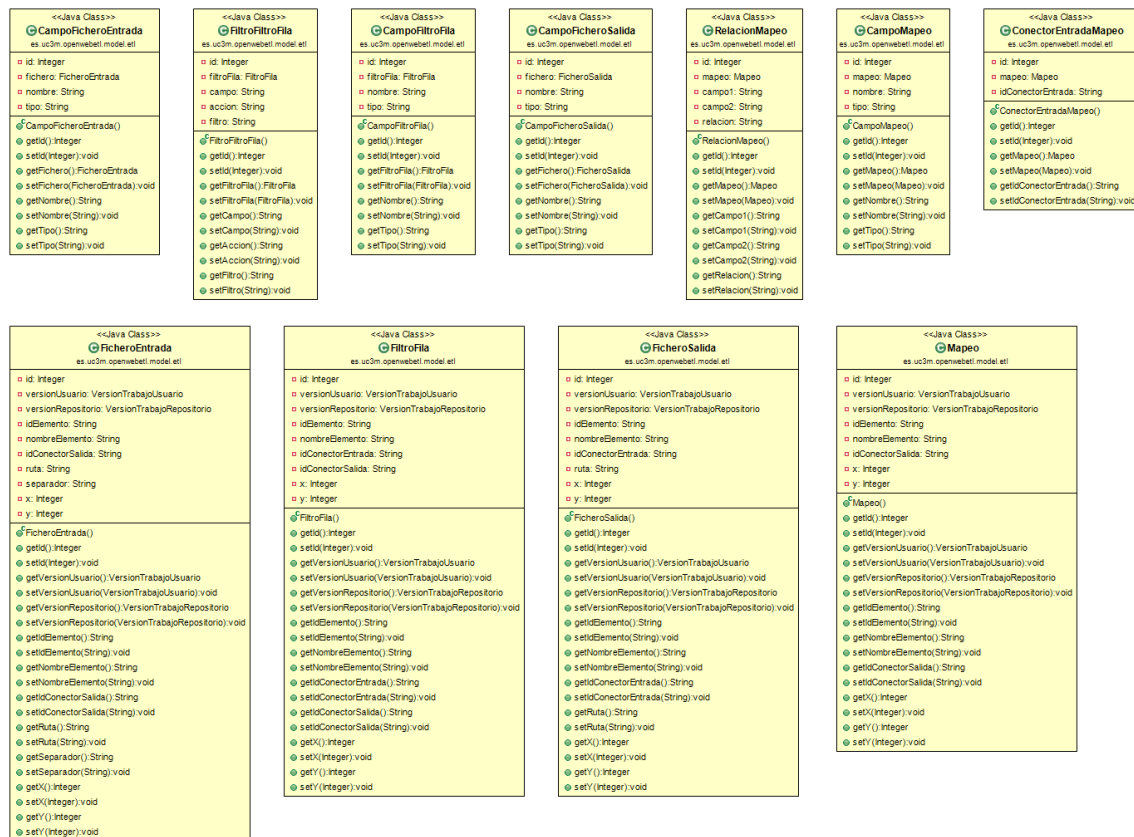


Ilustración 42 - Clases del Modelo parte ETL

4.2.3.3. Capa de negocio: Servicio

Esta capa de negocio será la encargada de leer y escribir de la base de datos de la aplicación, enviándole los resultados al Controlador. Es decir, el Controlador es el que pide al Servicio que haga una determinada acción sobre el medio físico en el que se guardan los datos.

La capa del Servicio estará representada por las clases ubicadas dentro del paquete 'es.uc3m.openwebetl.service'. Se presenta cada clase de este paquete con una descripción y el listado de los campos y métodos de cada una de ellas. El nombre de estos campos y métodos son descriptivos, por lo que no será necesario describirlos. Con toda esta información, se podrán implementar las clases de la capa de negocio del Servicio.

Antes de presentar las clases, se indican los siguientes aspectos que deben tenerse en cuenta para poder realizar la implementación con éxito:

Cada una de estas clases debe incluir la anotación `@Service` indicando el nombre del servicio con el que se conocerá. Con esta anotación se indica que la clase define un servicio.

La anotación `@Transactional` será empleada por todas las clases del paquete menos por `MailService.java`, ya que es la única que no trabajará con la base de datos. Con esta anotación se asegura que toda operación de la base de datos se realiza dentro de una transacción.

Todas las clases que trabajen con la base de datos incluirán el método `protected Session getSession()` que permitirá obtener la sesión actual de la clase `SessionFactory` indicada con la anotación `@Autowired`. Mediante esa sesión se podrá emplear las funciones de la API Criteria que se encuentra dentro de Hibernate. Esta API permitirá especificar consultas programáticamente (en base a clases y métodos de estas clases) sobre las entidades anteriormente explicadas definiendo un conjunto de restricciones. El equipo de trabajo que implemente la aplicación deberá saber cómo realizar operaciones sobre la base de datos, como inserción, borrado y actualización, utilizando Criteria.

Es importante señalar que cada método que realice una operación de inserción, modificación o borrado sobre alguna tabla de la base de datos deberá devolver un mensaje indicando si la operación se ha efectuado satisfactoriamente o no y una descripción de la misma.

Una vez dadas las pautas de cómo se deberán implementar este tipo de clases, se detalla cada una de las clases implicadas en esta capa de negocio:

- **AccesoService.java**

Mediante esta clase se podrá recuperar el identificador de la base de datos de un usuario a partir de su clave y el usuario entero a partir de su nombre. Es decir, se obtendrá la información necesaria para comprobar si un usuario consta en base de datos.

- **AdminService.java**

Esta clase permitirá realizar operaciones sobre la base de datos vinculadas con la administración de la aplicación y, más concretamente, con la gestión de áreas de trabajo y usuarios.

La información de base de datos con la que trabaja esta clase es restringida y sólo podrá ser accedida por un usuario con rol de Administrador. Por este motivo, para poder restringir el acceso a sus métodos, se empleará la anotación

`@PreAuthorize("hasRole('ROLE_ADMINISTRADOR')")` en cada uno de ellos. Con esta anotación se asegura que antes de acceder al método se pregunta por el rol del usuario que está intentando dicho acceso.

- **AreaTrabajoService.java**

Esta clase permitirá realizar operaciones sobre la base de datos vinculadas con la gestión de las áreas de trabajo.

La información de base de datos con la que trabaja esta clase es restringida y sólo podrá ser accedida por un usuario con rol de Administrador. Por este motivo, para poder restringir el acceso a sus métodos, se empleará la anotación `@PreAuthorize("hasRole('ROLE_USUARIO')")` en cada uno de ellos. Con esta anotación se asegura que antes de acceder al método se pregunta por el rol del usuario que está intentando dicho acceso.

- **ClaveUsuarioService.java**

Esta clase permitirá realizar operaciones sobre la base de datos vinculadas con la funcionalidad que implica el reinicio de la clave de un usuario.

- **CustomUserDetailsService.java**

Esta clase permitirá realizar operaciones sobre la base de datos vinculadas a la identificación de usuarios y sus permisos para el control del acceso a la aplicación.

- **EtlService.java**

Esta clase permitirá realizar operaciones sobre la base de datos vinculadas a la elaboración de proceso ETL.

La información de base de datos con la que trabaja esta clase es restringida y sólo podrá ser accedida por un usuario con rol de Usuario. Por este motivo, para poder restringir el acceso a sus métodos, se empleará la anotación `@PreAuthorize("hasRole('ROLE_USUARIO')")` en cada uno de ellos. Con esta anotación se asegura que antes de acceder al método se pregunta por el rol del usuario que está intentando dicho acceso.

- **MailService.java**

Esta clase permitirá enviar un correo electrónico y establecer el contenido del mismo haciendo uso de la librería JavaMail.

A continuación se ilustran las clases del Servicio con el detalle de sus campos y métodos. Aplicar zoom en la siguiente ilustración para poder visualizar la información correctamente:

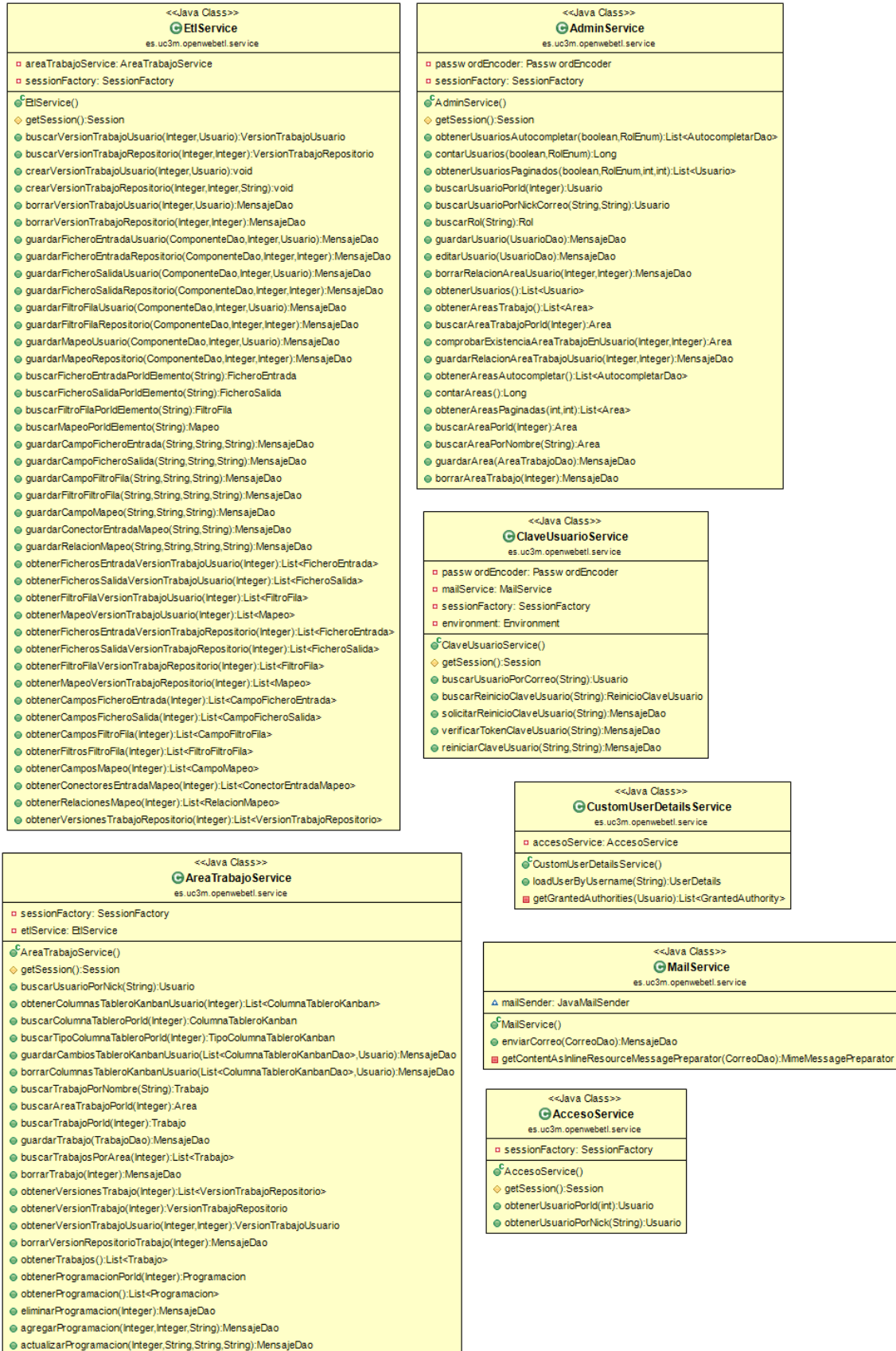


Ilustración 43 - Clases del Servicio

4.2.3.4. Capa de negocio: Controlador

El Controlador se encargará de procesar las peticiones del usuario, apoyándose en la capa de negocio de Servicio, y la construcción del modelo apropiado, pasándolo a la vista para la representación. Sobre esta capa de Servicio, el Controlador es el que pedirá a ésta que haga una determinada acción sobre el medio físico en el que se guardan los datos.

Esta capa de negocio estará representada por las clases ubicadas dentro del paquete 'es.uc3m.openwebetl.controller'. Deberán emplearse dos anotaciones fundamentales de Spring para indicar que estas clases serán controladores:

`@RestController` que permitirá indicar que una clase será un controlador que ofrecerá un servicio REST. Los métodos con esta anotación intercambiarán información con la parte del cliente en formato JSON gracias a la librería Jackson comentada en la fase del diseño. Cada método incluirá la anotación `@JsonView(Views.Public.class)`, para controlar lo que debe ser devuelto de nuevo a la solicitud. Para ello, será necesario crear la clase comodín 'Views.java' [86] dentro del paquete 'es.uc3m.openwebetl.jsonview'.

`@Controller` que permitirá indicar que una clase será un controlador estándar.

A continuación, se detallan las clases de esta capa de negocio:

- **AdminController.java**

Este controlador será un servicio REST cuyos métodos devolverán la información en formato JSON. Mediante la anotación `@RequestMapping` de Spring habrá que indicar que este controlador se ocupará de aquellas URLs que empiecen por "/admin".

Esta clase empleará el servicio que ofrece `AdminService.java`, ya que las operaciones que realizan los métodos de esta clase del Controlador requerirán a su vez aquellas acciones sobre la base de datos manejadas en esa clase del Servicio.

Esta clase del Controlador se encargará de procesar peticiones del usuario vinculadas a la administración, concretamente a la gestión de áreas de trabajo y usuarios. A continuación, se listan sus métodos incluyendo las anotaciones que tienen en su cabecera, de tal manera que gracias a los nombres descriptivos de estos métodos, sus anotaciones y parámetros, se podrán dilucidar las operaciones que deberán incluir.

Aplicar zoom en las siguientes ilustraciones para poder visualizar los métodos correctamente:


```

@RequestMapping(value = "/listado_usuarios_autocompletar/{esActivo}/{rol}", method = RequestMethod.GET)
public List<AutocompletarDao> obtenerListadoUsuariosAutocompletar(@PathVariable("esActivo") boolean esActivo, @PathVariable("rol") String rol)

@RequestMapping(value = "/listado_usuarios_paginados/{esActivo}/{rol}/{inicio}/{maximo}", method = RequestMethod.GET)
public List<UsuarioDao> obtenerListadoUsuariosPaginados(@PathVariable("esActivo") boolean esActivo,
@PathVariable("rol") String rol, @PathVariable("inicio") Integer inicio, @PathVariable("maximo") Integer maximo)

@RequestMapping(value = "/buscar_usuario_concreto/{idUsuario}", method = RequestMethod.GET)
public List<UsuarioDao> buscarUsuarioConcreto(@PathVariable("idUsuario") Integer idUsuario)

@RequestMapping(value = "/contador_usuarios/{esActivo}/{rol}", method = RequestMethod.GET)
public Long contarUsuarios(@PathVariable("esActivo") boolean esActivo, @PathVariable("rol") String rol)

@RequestMapping(value = "/guardar_usuario", method = RequestMethod.POST)
public MensajeDao guardarUsuario(@RequestBody UsuarioDao usuario)

@RequestMapping(value = "/editar_usuario", method = RequestMethod.POST)
public MensajeDao editarUsuario(@RequestBody UsuarioDao usuario)

@RequestMapping(value = "/buscar_areas_trabajo_usuario/{idUsuario}", method = RequestMethod.GET)
public List<AreaTrabajoDao> buscarAreasTrabajoUsuario(@PathVariable("idUsuario") Integer idUsuario)

@RequestMapping(value = "/borrado_relacion_area_usuario", method = RequestMethod.POST)
public MensajeDao borrarRelacionAreaUsuario(@RequestBody UsuarioAreaDao usuarioArea)

@RequestMapping(value = "/listado_areas_no_usuario_autocompletar/{idUsuario}", method = RequestMethod.GET)
public List<AutocompletarDao> obtenerListadoAreasNoUsuarioAutocompletar(@PathVariable("idUsuario") Integer idUsuario)

@RequestMapping(value = "/guardar_relacion_area_usuario", method = RequestMethod.POST)
public MensajeDao guardarRelacionAreaTrabajoUsuario(@RequestBody UsuarioAreaDao relacion)

```

Ilustración 44 - AdminController_parte1

```

@RequestMapping(value = "/listado_areas_autocompletar", method = RequestMethod.GET)
public List<AutocompletarDao> obtenerListadoAreasAutocompletar()

@RequestMapping(value = "/listado_areas_paginadas/{inicio}/{maximo}", method = RequestMethod.GET)
public List<AreaTrabajoDao> obtenerListadoAreasPaginadas(@PathVariable("inicio") Integer inicio, @PathVariable("maximo") Integer maximo)

@RequestMapping(value = "/buscar_area_concreta/{idArea}", method = RequestMethod.GET)
public List<AreaTrabajoDao> buscarAreaConcreta(@PathVariable("idArea") Integer idArea)

@RequestMapping(value = "/contador_areas", method = RequestMethod.GET)
public Long contarUsuarios()

@RequestMapping(value = "/guardar_area", method = RequestMethod.POST)
public MensajeDao guardarArea(@RequestBody AreaTrabajoDao area)

@RequestMapping(value = "/buscar_usuarios_area/{idArea}", method = RequestMethod.GET)
public List<UsuarioDao> buscarUsuariosAreas(@PathVariable("idArea") Integer idArea)
public List<UsuarioDao> buscarUsuariosAreas(@PathVariable("idArea") Integer idArea)

@RequestMapping(value = "/borrado_area", method = RequestMethod.POST)
public MensajeDao borrarAreaTrabajo(@RequestBody AreaTrabajoDao area)

@RequestMapping(value = "/listado_usuarios_no_area_autocompletar/{idArea}", method = RequestMethod.GET)
public List<AutocompletarDao> obtenerListadoUsuariosNoAreaAutocompletar(@PathVariable("idArea") Integer idArea)

```

Ilustración 45 - AdminController_parte2

- **AreaTrabajoController.java**

Este controlador será un servicio REST cuyos métodos devolverán la información en formato JSON. Mediante la anotación `@RequestMapping` de Spring habrá que indicar que este controlador se ocupará de aquellas URLs que empiecen por `"/área_trabajo"`.

Esta clase empleará el servicio que ofrece `AreaTrabajoService.java`, ya que las operaciones que realizan los métodos de esta clase del Controlador requerirán a su vez aquellas acciones sobre la base de datos manejadas en esa clase del Servicio.

Esta clase del Controlador se encargará de procesar peticiones del usuario vinculadas a la planificación de actividades de trabajo (tablero Kanban), gestión de áreas de trabajo y programación de trabajos. A continuación, se listan sus métodos incluyendo las anotaciones que tienen en su cabecera, de tal manera que gracias a los nombres descriptivos de estos métodos, sus anotaciones y parámetros, se podrán dilucidar las operaciones que deberán incluir.

Aplicar zoom en la siguiente ilustración para poder visualizar los métodos correctamente:

```
@RequestMapping(value = "/obtener_tablerokanban_usuario", method = RequestMethod.GET)
public List<ColumnaTableroKanbanDao> obtenerTableroKanbanUsuario(Authentication authentication)

@RequestMapping(value = "/guardar_cambios_tablerokanban_usuario", method = RequestMethod.POST)
public MensajeDao guardarCambiosTableroKanbanUsuario(@RequestBody ListadoColumnaTableroKanbanDao tablero,
    Authentication authentication)

@RequestMapping(value = "/obtener_areastrabajo_usuario", method = RequestMethod.GET)
public List<AutocompletarDao> obtenerAreasTrabajoUsuarioAutocompletar(Authentication authentication)

@RequestMapping(value = "/es_areastrabajo_usuario/{idArea}", method = RequestMethod.GET)
public AreaTrabajoDao esAreaTrabajoUsuario(@PathVariable("idArea") Integer idArea, Authentication authentication)

@RequestMapping(value = "/existe_trabajo_area", method = RequestMethod.POST)
public boolean existeTrabajoAreaPorNombre(@RequestBody TrabajoDao trabajo)

@RequestMapping(value = "/guardar_trabajo_area", method = RequestMethod.POST)
public MensajeDao guardarCambiosTableroKanbanUsuario(@RequestBody TrabajoDao trabajo)

@RequestMapping(value = "/obtener_trabajos_area/{idArea}", method = RequestMethod.GET)
public List<AutocompletarDao> obtenerTrabajosAreaAutocompletar(@PathVariable("idArea") Integer idArea)

@RequestMapping(value = "/obtener_datos_trabajos/{idTrabajo}", method = RequestMethod.GET)
public ContenedorInfoTrabajoDao obtenerDatosTrabajo(@PathVariable("idTrabajo") Integer idTrabajo,
    Authentication authentication)

@RequestMapping(value = "/borrar_trabajo", method = RequestMethod.POST)
public MensajeDao borrarTrabajo(@RequestBody TrabajoDao trabajo)

@RequestMapping(value = "/borrar_version_repositorio_trabajo", method = RequestMethod.POST)
public MensajeDao borrarVersionRepositorioTrabajo(@RequestBody VersionDao version)

@RequestMapping(value = "/obtener_versiones_repositorio_trabajo/{idTrabajo}", method = RequestMethod.GET)
public List<VersionDao> obtenerVersionesRepositorioTrabajo(@PathVariable("idTrabajo") Integer idTrabajo)

@RequestMapping(value = "/obtener_programaciones", method = RequestMethod.GET)
public List<ProgramacionDao> obtenerProgramaciones()

@RequestMapping(value = "/agregar_programacion", method = RequestMethod.POST)
public MensajeDao agregarProgramacion(@RequestBody ProgramacionDao programacion)

@RequestMapping(value = "/eliminar_programacion", method = RequestMethod.POST)
public MensajeDao eliminarProgramacion(@RequestBody ProgramacionDao programacion)

@RequestMapping(value = "/actualizar_programacion", method = RequestMethod.POST)
public MensajeDao actualizarProgramacion(@RequestBody ProgramacionDao programacion)

@RequestMapping(value = "/obtener_trabajos", method = RequestMethod.GET)
public List<TrabajoDao> obtenerTrabajos()
```

Ilustración 46 - AreaTrabajoController

- **ClaveUsuarioController.java**

Este controlador será un servicio REST cuyos métodos devolverán la información en formato JSON. Mediante la anotación `@RequestMapping` de Spring habrá que indicar que este controlador se ocupará de aquellas URLs que empiecen por `"/clave/rest"`.

Esta clase empleará el servicio que ofrece `ClaveUsuarioService.java`, ya que las operaciones que realizan los métodos de esta clase del Controlador requerirán a su vez aquellas acciones sobre la base de datos manejadas en esa clase del Servicio.

Esta clase del Controlador se encargará de procesar peticiones del usuario vinculadas a la recuperación de claves de usuario para el acceso a la aplicación. A continuación, se listan sus métodos incluyendo las anotaciones que tienen en su cabecera, de tal manera que gracias a los nombres descriptivos de estos métodos, sus anotaciones y parámetros, se podrán dilucidar las operaciones que deberán incluir.

Aplicar zoom en la siguiente ilustración para poder visualizar los métodos correctamente:


```
@RequestMapping(value = "/solicitud_reinicio_clave_usuario", method = RequestMethod.POST)
public MensajeDao solicitarReinicioClaveUsuario(@RequestBody UsuarioDao usuario)

@RequestMapping(value = "/verificacion_token_clave_usuario/{token}", method = RequestMethod.GET)
public MensajeDao verificarTokenClaveUsuario(@PathVariable("token") String token)

@RequestMapping(value = "/reinicio_clave_usuario", method = RequestMethod.POST)
public MensajeDao recordarClaveUsuario(@RequestBody UsuarioDao usuario)
```

Ilustración 47 - ClaveUsuarioController

- **ControlController.java**

Este controlador será un servicio REST cuyos métodos devolverán la información en formato JSON.

Esta clase del Controlador se encargará de procesar peticiones del usuario vinculadas a la obtención del rol del usuario. A continuación, se listan sus métodos incluyendo las anotaciones que tienen en su cabecera, de tal manera que gracias a los nombres descriptivos de estos métodos, sus anotaciones y parámetros, se podrán dilucidar las operaciones que deberán incluir.

Aplicar zoom en la siguiente ilustración para poder visualizar los métodos correctamente:

```
@RequestMapping(value = "/rol", method = RequestMethod.GET)
public CodigoPermisosDao obtenerCodigoPermisos(Authentication authentication)
```

Ilustración 48 – ControlController

- **EtlController.java**

Este controlador será un servicio REST cuyos métodos devolverán la información en formato JSON. Mediante la anotación `@RequestMapping` de Spring habrá que indicar que este controlador se ocupará de aquellas URLs que empiecen por `"/area_trabajo/etl"`.

Esta clase empleará el servicio que ofrece `AreaTrabajoService.java` y el que ofrece `EtlService.java`, ya que las operaciones que realizan los métodos de esta clase del Controlador requerirán a su vez aquellas acciones sobre la base de datos manejadas en esas clases del Servicio.

Esta clase del Controlador se encargará de procesar peticiones del usuario vinculadas a la elaboración de procesos ETL. A continuación, se listan sus métodos incluyendo las anotaciones que tienen en su cabecera, de tal manera que gracias a los nombres descriptivos de estos métodos, sus anotaciones y parámetros, se podrán dilucidar las operaciones que deberán incluir.

Aplicar zoom en las siguientes ilustraciones para poder visualizar los métodos correctamente:

```
@RequestMapping(value = "/recrear_version_trabajo_usuario", method = RequestMethod.POST)
public void recrearVersionTrabajoUsuario(@RequestBody TrabajoDao trabajo, Authentication authentication)

@RequestMapping(value = "/recrear_version_trabajo_repositorio", method = RequestMethod.POST)
public void recrearVersionTrabajoRepositorio(@RequestBody VersionDao version)

@RequestMapping(value = "/guardar_ficheroentrada_usuario/{idTrabajo}", method = RequestMethod.POST)
public MensajeDao guardarFicheroEntradaUsuario(@RequestBody ComponenteDao componente, Authentication authentication,
    @PathVariable("idTrabajo") Integer idTrabajo)

@RequestMapping(value = "/guardar_ficheroentrada_repositorio/{idTrabajo}/{numversion}", method = RequestMethod.POST)
public MensajeDao guardarFicheroEntradaRepositorio(@RequestBody ComponenteDao componente,
    @PathVariable("idTrabajo") Integer idTrabajo, @PathVariable("numversion") Integer numversion)

@RequestMapping(value = "/guardar_ficherosalida_usuario/{idTrabajo}", method = RequestMethod.POST)
public MensajeDao guardarFicheroSalidaUsuario(@RequestBody ComponenteDao componente, Authentication authentication,
    @PathVariable("idTrabajo") Integer idTrabajo)

@RequestMapping(value = "/guardar_ficherosalida_repositorio/{idTrabajo}/{numversion}", method = RequestMethod.POST)
public MensajeDao guardarFicheroSalidaRepositorio(@RequestBody ComponenteDao componente,
    @PathVariable("idTrabajo") Integer idTrabajo, @PathVariable("numversion") Integer numversion)

@RequestMapping(value = "/guardar_filtrofila_usuario/{idTrabajo}", method = RequestMethod.POST)
public MensajeDao guardarFiltroFilaUsuario(@RequestBody ComponenteDao componente, Authentication authentication,
    @PathVariable("idTrabajo") Integer idTrabajo)

@RequestMapping(value = "/guardar_filtrofila_repositorio/{idTrabajo}/{numversion}", method = RequestMethod.POST)
public MensajeDao guardarFiltroFilaUsuario(@RequestBody ComponenteDao componente,
    @PathVariable("idTrabajo") Integer idTrabajo, @PathVariable("numversion") Integer numversion)

@RequestMapping(value = "/guardar_mapeo_usuario/{idTrabajo}", method = RequestMethod.POST)
public MensajeDao guardarMapeoUsuario(@RequestBody ComponenteDao componente, Authentication authentication,
    @PathVariable("idTrabajo") Integer idTrabajo)

@RequestMapping(value = "/guardar_mapeo_repositorio/{idTrabajo}/{numversion}", method = RequestMethod.POST)
public MensajeDao guardarMapeoUsuario(@RequestBody ComponenteDao componente,
    @PathVariable("idTrabajo") Integer idTrabajo, @PathVariable("numversion") Integer numversion)

@RequestMapping(value = "/guardar_campo_ficheroentrada/{idElementoFicheroEntrada}", method = RequestMethod.POST)
public MensajeDao guardarCampoFicheroEntrada(@RequestBody CampoDao campo, Authentication authentication,
    @PathVariable("idElementoFicheroEntrada") String idElementoFicheroEntrada)

@RequestMapping(value = "/guardar_campo_ficherosalida/{idElementoFicheroSalida}", method = RequestMethod.POST)
public MensajeDao guardarCampoFicheroSalida(@RequestBody CampoDao campo, Authentication authentication,
    @PathVariable("idElementoFicheroSalida") String idElementoFicheroSalida)

@RequestMapping(value = "/guardar_campo_filtrofila/{idElementoFiltroFila}", method = RequestMethod.POST)
public MensajeDao guardarCampoFiltroFila(@RequestBody CampoDao campo, Authentication authentication,
    @PathVariable("idElementoFiltroFila") String idElementoFiltroFila)
```

Ilustración 49 - EtlController_parte1

```

@RequestMapping(value = "/guardar_filtro_filtrofila/{idElementoFiltroFila}", method = RequestMethod.POST)
public MensajeDao guardarCampoFiltroFila(@RequestBody FiltroDao filtro, Authentication authentication,
    @PathVariable("idElementoFiltroFila") String idElementoFiltroFila)

@RequestMapping(value = "/guardar_campo_mapeo/{idElementoMapeo}", method = RequestMethod.POST)
public MensajeDao guardarCampoMapeo(@RequestBody CampoDao campo, Authentication authentication,
    @PathVariable("idElementoMapeo") String idElementoMapeo)

@RequestMapping(value = "/guardar_relacion_mapeo/{idElementoMapeo}", method = RequestMethod.POST)
public MensajeDao guardarRelacionMapeo(@RequestBody RelacionDao relacion, Authentication authentication,
    @PathVariable("idElementoMapeo") String idElementoMapeo)

@RequestMapping(value = "/guardar_conectorentrada_mapeo/{idElementoMapeo}", method = RequestMethod.POST)
public MensajeDao guardarConectorEntradaMapeo(@RequestBody ConectorEntradaMapeoDao conector,
    Authentication authentication, @PathVariable("idElementoMapeo") String idElementoMapeo)

private void obtenerFicherosEntradaVersionUsuario(Usuario usuario, Integer idTrabajo,
    List<ComponenteDao> resultado)

private void obtenerFicherosEntradaVersionRepositorio(Integer numversion, Integer idTrabajo,
    List<ComponenteDao> resultado)

private void obtenerFicherosSalidaVersionUsuario(Usuario usuario, Integer idTrabajo,
    List<ComponenteDao> resultado)

private void obtenerFicherosSalidaVersionRepositorio(Integer numversion, Integer idTrabajo,
    List<ComponenteDao> resultado)

private void obtenerFiltroFilaVersionUsuario(Usuario usuario, Integer idTrabajo, List<ComponenteDao> resultado)

private void obtenerFiltroFilaVersionRepositorio(Integer numversion, Integer idTrabajo,
    List<ComponenteDao> resultado)

private void obtenerMapeoVersionUsuario(Usuario usuario, Integer idTrabajo, List<ComponenteDao> resultado)

private void obtenerMapeoVersionRepositorio(Integer numversion, Integer idTrabajo, List<ComponenteDao> resultado)

@RequestMapping(value = "/obtener_version_trabajo_usuario/{idTrabajo}", method = RequestMethod.GET)
public List<ComponenteDao> obtenerVersionTrabajoUsuario(@PathVariable("idTrabajo") Integer idTrabajo,
    Authentication authentication)

@RequestMapping(value = "/obtener_version_trabajo_repositorio/{idTrabajo}/{numversion}", method = RequestMethod.GET)
public List<ComponenteDao> obtenerVersionTrabajoRepositorio(@PathVariable("idTrabajo") Integer idTrabajo,
    @PathVariable("numversion") Integer numversion)

@RequestMapping(value = "/obtener_versiones_trabajo_repositorio/{idTrabajo}", method = RequestMethod.GET)
public List<VersionDao> obtenerVersionesTrabajoRepositorio(@PathVariable("idTrabajo") Integer idTrabajo)

```

Ilustración 50 - EtlController_parte2

La implementación de la elaboración de procesos ETL requiere una explicación a fondo, pero a los controladores de esta clase les viene el trabajo hecho. Es en uno de los controladores de la parte del cliente donde se encuentra el motor de esta funcionalidad. Por tanto, se detallarán las estructuras y operaciones de este controlador en el apartado de la vista para que a partir de esta información se pueda implementar esta funcionalidad.

- **MainController.java**

Esta clase del Controlador se encargará de mostrar al usuario las páginas de la aplicación por medio de las URLs a las que acceda, cargando las vistas requeridas por cada una de ellas. A continuación, se listan sus métodos incluyendo las anotaciones que tienen en su cabecera, de tal manera que gracias a los nombres descriptivos de estos métodos, sus anotaciones y parámetros, se podrán dilucidar las operaciones que deberán incluir.

Aplicar zoom en la siguiente ilustración para poder visualizar los métodos correctamente:

```
@RequestMapping(value = "/", method = RequestMethod.GET)
public String homePage(ModelMap model)

@RequestMapping(value = "/main", method = RequestMethod.GET)
public String adminPage(ModelMap model)

@RequestMapping(value = "/login", method = RequestMethod.GET)
public String loginPage()

@RequestMapping(value = "/clave/solicitud_reinicio_clave_usuario", method = RequestMethod.GET)
public String solicitudReinicioClaveUsuario()

@RequestMapping(value = "/clave/reinicio_clave_usuario/{token}", method = RequestMethod.GET)
public String reinicioClaveUsuario(ModelMap model, @PathVariable("token") String token)

@RequestMapping(value = "/logout", method = RequestMethod.GET)
public String logoutPage(HttpServletRequest request, HttpServletResponse response)
```

Ilustración 51 - MainController

Antes de concluir con el apartado del Controlador, es importante mencionar que se necesitarán una serie de clases que deben ser incluidas dentro del paquete 'es.uc3m.openwebetl.dao'. Estas clases permitirán el intercambio de información en formato JSON entre el cliente y los controladores.

A continuación se ilustran estas clases con el detalle de sus campos y métodos. Aplicar zoom en las siguientes ilustraciones para poder visualizar la información correctamente:

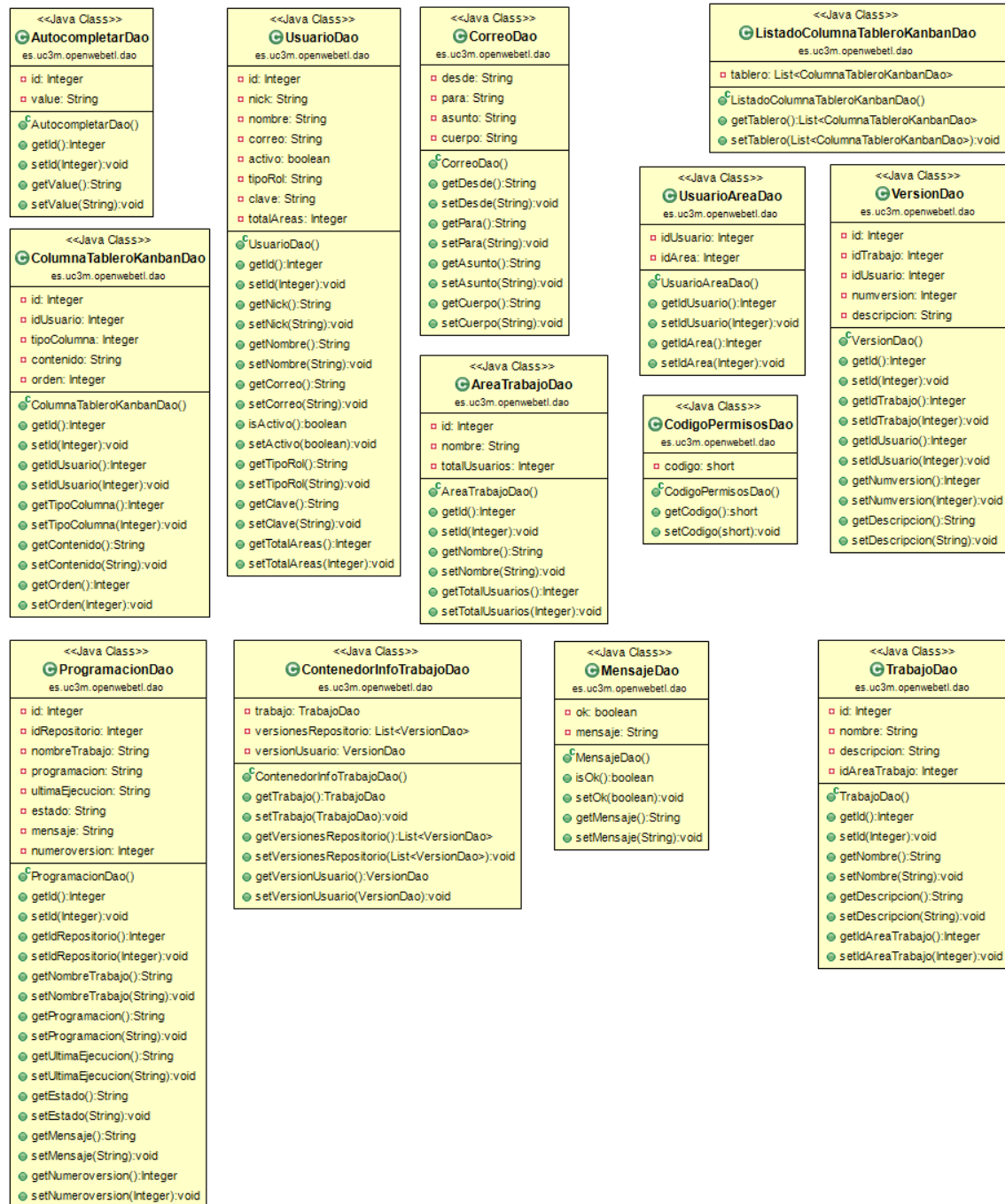


Ilustración 52 - Clases dao

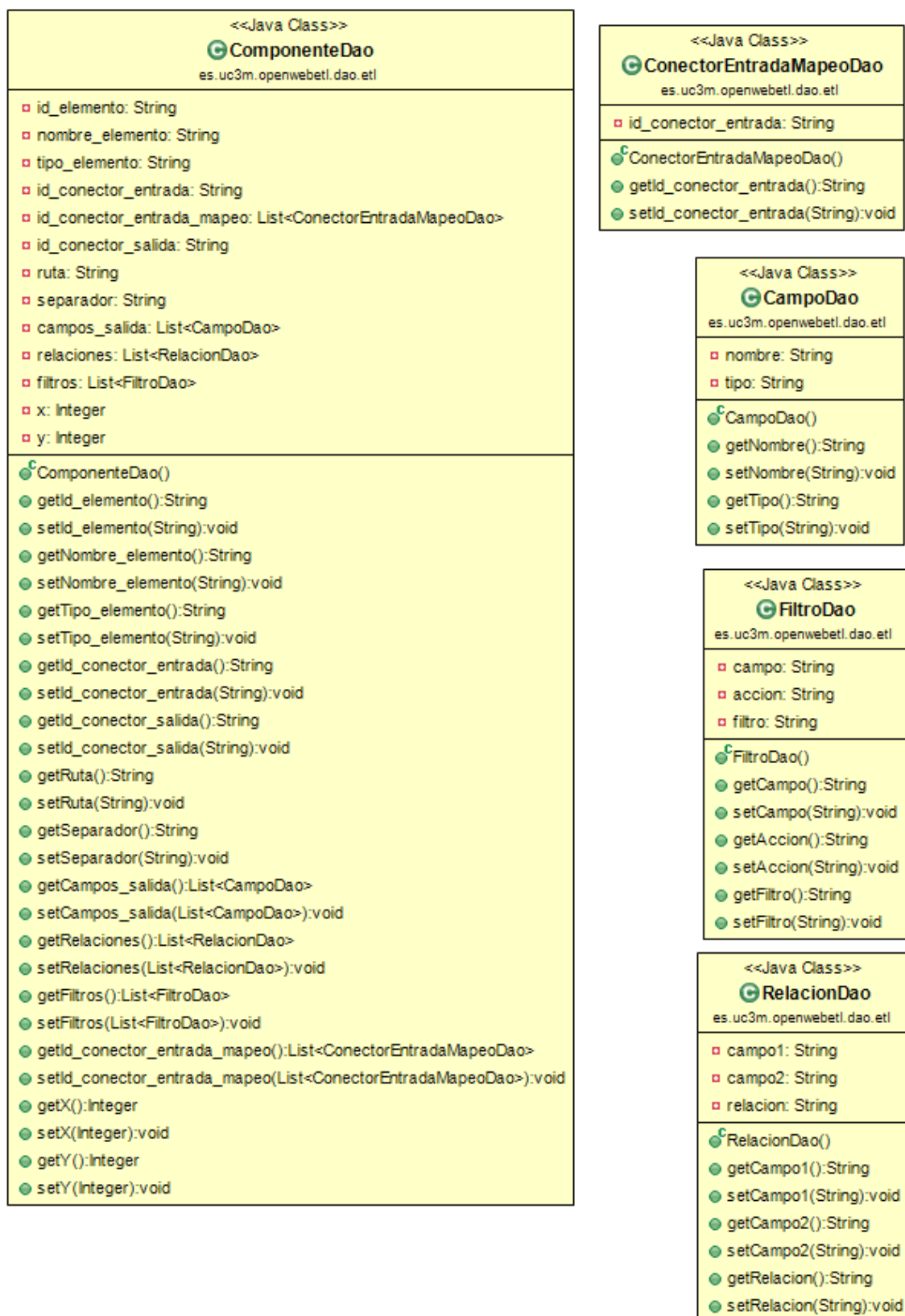


Ilustración 53 - Clases dao etl

4.2.3.5. Capa de negocio: Vista

Las capas de negocio que se han visto hasta el momento pertenecen a la parte del servidor. Sin embargo, esta capa que aquí se presenta pertenece a la parte del cliente. La vista presenta la información y lógica de negocio en un formato adecuado para interactuar con la interfaz del usuario. Se encargará de enviar las peticiones del usuario a la parte del servidor (a través del Controlador) principalmente mediante el intercambio de información en formato JSON. Asimismo, mostrará al usuario la información recibida por parte del servidor.

Las vistas serán documentos JSP que estarán ubicados en el directorio `'/src/webapp/WEB-INF/views'`. Estas vistas serán las siguientes:

- **acceso.jsp**: se corresponde con la interfaz de acceso a la aplicación.
- **reinicioClaveUsuario.jsp**: se corresponde con la interfaz desde la que el usuario cambiará su clave.
- **solicitudReinicioClaveUsuario.jsp**: se corresponde con la interfaz desde la que el usuario solicitará un reinicio de clave de usuario.
- **main.jsp**: esta vista será la base de la aplicación de tipo SPA una vez que un usuario acreditado acceda al contenido de la aplicación. En ella se cargarán los archivos JavaScript y CSS necesarios para poder trabajar con las librerías de Bootstrap, Angularjs, jQuery y jsPlumb.

Como ya se ha mencionado, una de las librerías cargadas en `main.jsp` será Angularjs que trae consigo el módulo `ngRoute` que ayudará a convertir la aplicación en SPA. Este módulo permitirá tener una aplicación de una única página a partir de `main.jsp`, pero representando URLs distintas, simulando una navegación a través de la aplicación. Para ello, habrá que especificar una URL que se corresponderá con un enlace interno, la plantilla HTML a mostrar y el controlador con las funciones JavaScript que tendrá asociadas. En la única página real que se tenga (`main.jsp`) se deberá especificar con una etiqueta especial de Angularjs el lugar donde la plantilla HTML se mostrará al acceder a la ruta correspondiente al enlace interno definido anteriormente.

A continuación, se muestra la correspondencia entre enlaces internos, plantillas HTML y controladores que deben existir:

Enlace interno	Plantilla HTML	Controlador
'/'	control.html	controlCtrl
'/admin/usuario'	adminUsuario.html	adminUsuarioCtrl
'/admin/area'	adminAreaTrabajo.html	adminAreaTrabajoCtrl
'/area_trabajo/tableroKanban'	tableroKanban.html	areaTrabajoTableroKanbanCtrl
'/area_trabajo/menuAreasTrabajo'	menuAreasTrabajo.html	menuAreasTrabajoCtrl
'/area_trabajo/menuTrabajosAreaTrabajo/:ID'	menuTrabajosAreaTrabajo.html	menuTrabajosAreaTrabajoCtrl

'/area_trabajo/trabajo/:	'/openwebetl/html/area_trabajo/	trabajoCtrl
TIPOVERSION/:AREAID/:	trabajo.html'	
TRABAJOID/:NUMVERSION'		
'/area_trabajo/programa	programacion.html	programacionCtrl
cion'		

Tabla 24 - Relación enlace-plantilla-controlador

Estas plantillas HTML se encontrarán en el directorio de la aplicación '/src/webapp/html'. A su vez, dentro del directorio 'admin' se encontrarán las plantillas de administración y dentro del directorio 'area_trabajo' las correspondientes al trabajo en sí. En el [apartado del diseño](#) se mostraron las interfaces de la aplicación y, puesto que los elementos de diseño empleados son los que ofrece la librería Bootstrap, no será necesario dar detalles sobre la implementación de las mismas.

Las plantillas HTML de administración serán las siguientes:

- **adminAreaTrabajo.html**: interfaz de la administración de áreas de trabajo.
- **adminUsuario.html**: interfaz de la administración de usuarios.

Las plantillas HTML de trabajo serán las siguientes:

- **menuAreasTrabajo.html**: interfaz de selección de áreas de trabajo.
- **menuTrabajosAreaTrabajo.html**: interfaz de un área de trabajo.
- **programacion.html**: interfaz de la programación de trabajos.
- **tableroKanban.html**: interfaz del tablero de planificación de tareas.
- **trabajo.html**: interfaz del área de elaboración de un proceso ETL.

Existirá una última plantilla HTML llamada `control.html` cuyo único objetivo será ser cargada mientras se verifican las credenciales de un usuario que intenta acceder a la aplicación.

Por su parte, los controladores estarán contenidos en archivos JavaScript que estarán cargados en `main.jsp`. Éstos se encargarán de hacer las llamadas al API REST y manejar la respuesta en formato JSON. Mediante los servicios `$http.post` y `$http.get` de Angularjs se realizará la comunicación con el servidor. `$http.post` se empleará cuando la operación que se solicite implique cambios en la base de datos y la información vendrá contenida dentro del cuerpo de la petición, para el resto de los casos se empleará `$http.get` y la información se pasará en forma de parámetros a través de las URLs.

Una vez que se conoce la forma en la que habrá que comunicarse con el servidor y habiendo visto en el diseño detallado anteriormente qué acciones se realizan en cada una de las vistas a las que están asociados estos controladores, no será necesario detallar la implementación de los mismos para poder llevarla a cabo. Excepto en el caso del controlador que se ocupa de la elaboración de un proceso ETL, en cuyo caso será necesario hacer hincapié en la lógica que conlleva para poder realizar su implementación.

Estos controladores se encontrarán dentro del directorio '/src/webapp/resources/js/controller' y serán los siguientes:

- **adminAreaTrabajoCtrl:** estará implementado en el archivo 'adminAreaTrabajoCtrl.js'. Se encargará de gestionar las solicitudes del cliente y las respuestas del servidor vinculadas a la interfaz de administración de áreas de trabajo.
- **adminUsuarioCtrl:** estará implementado en el archivo 'adminUsuarioCtrl.js'. Se encargará de gestionar las solicitudes del cliente y las respuestas del servidor vinculadas a la interfaz de administración de usuarios.
- **areaTrabajoTableroKanbanCtrl:** estará implementado en el archivo 'areaTrabajoTableroKanbanCtrl.js'. Se encargará de gestionar las solicitudes del cliente y las respuestas del servidor vinculadas a la interfaz del tablero de planificación de tareas de trabajo del usuario.
- **controlCtrl:** estará implementado en el archivo 'controlCtrl.js'. Redirige al usuario a la página de administración o a la de trabajo, dependiendo de si el usuario tiene rol de Administrador o de Usuario, respectivamente.
- **menuAreasTrabajoCtrl:** estará implementado en el archivo 'menuAreasTrabajoCtrl.js'. Se encargará de gestionar las solicitudes del cliente y las respuestas del servidor vinculadas a la interfaz selección de áreas de trabajo.
- **menuTrabajosAreaTrabajoCtrl:** estará implementado en el archivo 'menuTrabajosAreaTrabajoCtrl.js'. Se encargará de gestionar las solicitudes del cliente y las respuestas del servidor vinculadas a la interfaz de un área de trabajo y sus trabajos.
- **programacionCtrl:** estará implementado en el archivo 'programacionCtrl.js'. Se encargará de gestionar las solicitudes del cliente y las respuestas del servidor vinculadas a la interfaz de programación de trabajos.
- **reinicioClaveUsuarioCtrl:** estará implementado en el archivo 'reinicioClaveUsuario.js'. Se encargará de gestionar las solicitudes del cliente y las respuestas del servidor vinculadas al reinicio de claves de usuario.
- **solicitudReinicioClaveUsuarioCtrl:** estará implementado en el archivo 'solicitudReinicioClaveUsuario.js'. Se encargará de gestionar las solicitudes del cliente y las respuestas del servidor vinculadas a la solicitud de reinicio de claves de usuario.
- **trabajoCtrl:** estará implementado en el archivo 'trabajoCtrl.js'. Se encargará de gestionar las solicitudes del cliente y las respuestas del servidor vinculadas a la elaboración de procesos ETL.

4.2.3.6. Implementación del controlador para la elaboración de procesos ETL

Para poder mantener en memoria la información de cada componente ETL y las relaciones existentes entre ellos será necesario disponer de una matriz de elementos que representen componentes ETL. Cada elemento dentro de esta matriz contará con los siguientes campos:

- **Id_elemento:** será el identificador del componente.
- **Nombre_elemento:** será el nombre con el que el usuario identificará al componente en pantalla.
- **Tipo_elemento:** indicará el tipo de componente. Es decir, 'ficheroEntrada', 'ficheroSalida', 'mapeo' o 'filtroFila'.
- **Id_conector_entrada:** identificador del componente conectado con el componente actual a su entrada. El componente de tipo 'FicheroEntrada' será el único que tenga este campo nulo.

- **Id_conector_salida:** identificador del componente conectado con el componente actual a su salida. El componente de tipo 'FicheroSalida' será el único que tenga este campo nulo.
- **Filtros:** será una matriz de filtros que contendrá filtros cuyos campos serán el nombre del campo filtrado, el tipo de filtro y el valor a comparar. Este campo será utilizado sólo por el componente 'FiltroFila', el resto tendrá este campo nulo.
- **Campos_salida:** será una matriz de campos que contendrá campos cuyos campos serán el nombre del campo y el tipo de dato.
- **Ruta:** indicará la ruta de un fichero. Este campo será utilizado sólo por el componente 'FicheroEntrada' para indicar la ruta de su fichero de entrada y por el componente 'FicheroSalida' para indicar la ruta donde se creará el fichero de salida, el resto tendrá este campo nulo.
- **Separador:** indicará el carácter que separa las columnas de un fichero de entrada de tipo CSV. Este campo será utilizado sólo por el componente 'FicheroEntrada', el resto tendrá este campo nulo.
- **Relaciones:** será una matriz de relaciones que contendrá relaciones cuyos campos serán el nombre del campo del primer fichero de entrada, el nombre del campo del segundo fichero de entrada y el tipo de relación. Este campo será utilizado sólo por el componente 'Mapeo', el resto tendrá este campo nulo.

Por último, una vez vista la estructura de componentes, se presentan las funciones principales que deberán incluirse en el controlador 'trabajoCtrl'. Será necesario emplear la librería de JavaScript llamada jsPlumb para poder representar en la vista los componentes y los conectores que los unen. Para saber cómo representar en la vista estos componentes y conectores y, al mismo tiempo, saber cómo eliminarlos y moverlos por la pantalla empleando esta librería, se tendrá que estudiar su documentación [88].

- **Conectar componentes:**

Se deberá crear una función que reciba el identificador del componente de salida de los datos y el identificador del componente de destino de los datos. Buscará el elemento del componente de salida de los datos en la matriz de componentes por su identificador y comprobará si el campo del identificador del conector de salida es nulo. En caso contrario, buscará ese componente al que está conectado en la matriz de componentes y pondrá a nulo el valor del campo del identificador de conector de entrada. A continuación, en el campo del identificador del conector de salida del componente de salida de los datos se escribirá el identificador del componente de destino de los datos.

Seguidamente, buscará el elemento del componente de destino de los datos en la matriz de componentes por su identificador y comprobará si el campo del identificador del conector de entrada es nulo. En caso contrario, buscará ese componente al que está conectado en la matriz de componentes y pondrá a nulo el valor del campo del identificador de conector de salida. A continuación, en el campo del identificador del conector de entrada del componente de destino de los datos se escribirá el identificador del componente de salida de los datos.

Mediante las funciones de jsPlumb se pintará el conector que unirá ambos componentes en la vista.

- **Crear componente:**

Se deberá crear una función que reciba el tipo de componente a crear. Creará un elemento que representará un componente para posteriormente añadirlo a la matriz de componentes. Antes, añadirá a este elemento un identificador único a partir del contador de componentes existentes, un nombre basado en el tipo de componente y en el contador de componentes existentes y el tipo de elemento (componente) a crear.

Mediante las funciones de jsPlumb se pintará el nuevo componente en la vista.

- **Eliminar componente**

Se deberá crear una función que reciba el identificador del componente que se desea eliminar. Buscará el elemento del componente en la matriz de componentes y verificará que tengan valores nulos el campo del identificador del conector de entrada y el campo identificador del conector de salida. En caso contrario, no dejará eliminar el componente. Si fueran nulos, se obtendría la posición que ocupa este elemento en la matriz de componentes y se eliminaría de la misma.

Mediante las funciones de jsPlumb se eliminará de la vista la representación del componente eliminado.

- **Guardar campo mapeado**

Se deberá crear una función que reciba el identificador de un componente, el nombre del campo y el tipo de dato de éste. Creará una matriz de dos posiciones donde introducirá el nombre del campo y su tipo. Buscará el elemento por su identificador en la matriz de componentes y agregará la matriz del campo creada a la matriz de campos del componente en cuestión. Este cambio implicará actualizar las matrices de campos de todos los componentes de la matriz que estén conectados a partir de la salida de este componente.

- **Eliminar campo mapeado**

Se deberá crear una función que reciba el identificador de un componente y el nombre del campo a borrar. Buscará el elemento por su identificador en la matriz de componentes. Una vez localizado, buscará en su matriz de campos el campo a borrar a través de su nombre. Una vez localizada la posición del mismo, se eliminará de la matriz de campos. Este cambio implicará actualizar las matrices de campos de todos los componentes de la matriz que estén conectados a partir de la salida de este componente.

- **Guardar relación de campos**

Se deberá crear una función que reciba el identificador de un componente de tipo 'Mapeo', el nombre del campo del primer fichero, el nombre del campo del segundo fichero y el tipo de relación. Creará una matriz de tres posiciones donde insertará el nombre del campo del primer fichero, el nombre del campo del segundo fichero y el tipo de relación. Buscará el elemento por su identificador en la matriz de componentes y agregará la matriz de relación creada a la matriz de relaciones del componente en cuestión.

- **Eliminar relación de campos**

Se deberá crear una función que reciba el identificador de un componente de tipo 'Mapeo', el nombre del campo del primer fichero relacionado y el campo del segundo fichero relacionado

cuya relación se quiera borrar. Buscará el elemento por su identificador en la matriz de componentes. Una vez localizado, buscará en su matriz de relaciones la relación a borrar a partir del nombre del campo del primer fichero relacionado y del campo del segundo fichero relacionado. Una vez localizada la posición de la misma, se eliminará de la matriz de relaciones.

- **Guardar filtro**

Se deberá crear una función que reciba el identificador de un componente de tipo 'FiltroFila', el nombre del campo a filtrar, la condición del filtro y el valor con el que realizar la comparación. Creará una matriz de tres posiciones donde introducirá el nombre del campo a filtrar, la condición del filtro y el valor con el que realizar la comparación. Buscará el elemento por su identificador en la matriz de componentes y agregará la matriz del filtro creada a la matriz de filtros del componente en cuestión.

- **Eliminar filtro**

Se deberá crear una función que reciba el identificador de un componente de tipo 'FiltroFila' y el nombre del campo cuyo filtro se quiera eliminar. Buscará el elemento por su identificador en la matriz de componentes. Una vez localizado, buscará en su matriz de filtros el filtro a borrar a través del nombre del campo. Una vez localizada la posición del mismo, se eliminará de la matriz de filtros.

- **Guardar ruta y separador de columnas de fichero de entrada**

Se deberá crear una función que reciba el identificador de un componente de tipo 'FicheroEntrada', la ruta del fichero de entrada y el carácter separador de columnas del fichero. Buscará el elemento por su identificador en la matriz de componentes y agregará, por un lado, la ruta en el campo de ruta del elemento y, por otro, el separador en el campo separador de este elemento.

- **Guardar ruta de fichero de salida**

Se deberá crear una función que reciba el identificador de un componente de tipo 'FicheroSalida' y la ruta del fichero de salida. Buscará el elemento por su identificador en la matriz de componentes y agregará la ruta en el campo de ruta del elemento.

- **Verificar proceso ETL**

Se deberá crear una función que compruebe la consistencia del proceso ETL. Para ello, recorrerá todos los elementos de la matriz de componentes en busca de los siguientes errores:

Si el componente no es de tipo 'FicheroSalida' y no tiene un identificador de conector de salida, se informará de este error.

Si el componente no es de tipo 'FicheroEntrada' y no tiene un identificador de conector de entrada, se informará de este error.

Si el componente es de tipo 'FicheroEntrada' o 'FicheroSalida' y no tiene ruta, se informará de este error.

Si el componente es de tipo 'FicheroEntrada' y no tiene separador establecido, se informará de este error.

Si el componente tiene vacía su matriz de campos, se informará de este error.

- **Guardar trabajo con el proceso ETL**

Se deberá poder enviar una petición del cliente al servidor para guardar el proceso ETL elaborado en un trabajo. Para ello, se necesitará saber si se trata de una copia de usuario o de una versión de repositorio. En este segundo caso, se necesitará recibir el número de versión y la descripción. Además, se deberá enviar en formato JSON al controlador del servidor en cuestión, la matriz de componentes guardada en memoria con el objetivo que pueda ser desmontado por partes (componentes, campos, relaciones, filtros, conectores) para que pueda ser persistido en base de datos según la estructura de tablas del esquema de la aplicación atendiendo al modelo Entidad/Relación mostrado en el [apartado del diseño](#).

4.2.3.7. Despliegue de la aplicación

Una vez finalizada la implementación de la aplicación será necesario subirla al contenedor de aplicaciones del servidor de destino. Este servidor tendrá que tener instalado JRE 1.7 y JDK 7. Además, deberá tener instalado y levantado Apache Tomcat 8, contenedor de servlets donde se desplegará la aplicación web.

Será necesario generar un WAR de la aplicación desde la IDE de Eclipse. Para conseguir este paso, habrá que hacer clic con el botón derecho en el nombre del proyecto de la aplicación. Y, en el menú que aparecerá, seguir la siguiente ruta: 'Run as -> Maven clean'. Si la salida por consola indica que la ejecución ha sido correcta, entonces se repetirán los mismos pasos, pero en lugar de hacer clic en 'Maven clean', se hará en 'Maven install'. En la consola de ejecución de Eclipse, si no se han producido errores, se mostrará la ruta donde se haya generado el WAR de la aplicación.

El último paso será subir el WAR generado al Tomcat del servidor. Para ello, habrá que acceder a la interfaz de administración de este servidor de aplicaciones a través del navegador. Se seleccionará el WAR de la aplicación dentro del directorio de archivos del ordenador local y se hará clic en el botón de despliegue. Una vez finalizada la subida, se mostrará un mensaje informativo para saber si la operación se ha efectuado correctamente. Si éste fuera el caso, la aplicación ya estaría operativa.

Es importante que la base de datos de la aplicación esté operativa para que ésta funcione correctamente. Además, el esquema de la aplicación deberá tener cargado al menos un usuario con rol de Administrador, puesto que de otra forma no se podrá crear nuevos usuarios.

4.2.4. Pruebas

4.2.4.1. Pruebas de rendimiento

Confrontar el rendimiento de las ejecuciones de los procesos ETL generados con la herramienta ETL desarrollada con el de las ejecuciones de los procesos ETL generados con las herramientas ETL existentes en el mercado no es un objetivo que forme parte de este trabajo. Por este motivo, no se hará hincapié en las pruebas de rendimiento. No obstante, las pruebas realizadas al respecto determinaron que la ejecución de un mismo trabajo con la herramienta ETL desarrollada en este proyecto es más lenta que la realizada con una de las herramientas ETL más punteras del mercado.

4.2.4.2. Pruebas unitarias

Por otro lado, se han ido realizando pruebas unitarias tanto en los métodos de los controladores como en los métodos de los servicios a lo largo de toda la implementación de la aplicación. Una vez que se realizaba uno de estos métodos, se implementaba una prueba unitaria para probar su eficacia y, en caso de no superar alguna de ellas, se modificaba el método hasta que la superase. Para poder llevar a cabo estas pruebas se ha empleado el framework de Java llamado JUnit.

4.2.4.3. Pruebas funcionales

Por último, se han realizado las pruebas funcionales que se describieron en el apartado [“3.5. Plan de pruebas”](#). El resultado de estas pruebas, una vez finalizada la implementación de la aplicación, se va a mostrar a continuación utilizando el identificador de cada una de ellas establecido en ese apartado del plan de pruebas (para ver los pasos seguidos en cada prueba se deberá consultar ese apartado):

- **Pruebas de planificación de actividades de trabajo:**

Estas pruebas están destinadas a comprobar el correcto funcionamiento del módulo destinado al tablero de planificación de actividades de trabajo del usuario de la aplicación.

ID	P 1-1	Fecha	15/09/16	Resultado	Satisfactorio
Finalidad		Acceder a módulo de planificación de actividades de trabajo.			
ID	P 1-2	Fecha	15/09/16	Resultado	Satisfactorio
Finalidad		Visualizar tablero de planificación de trabajo.			
ID	P 1-3	Fecha	15/09/16	Resultado	Satisfactorio
Finalidad		Añadir etiqueta de trabajo.			
ID	P 1-4	Fecha	15/09/16	Resultado	Satisfactorio
Finalidad		Eliminar etiqueta de trabajo.			
ID	P 1-5	Fecha	15/09/16	Resultado	Satisfactorio
Finalidad		Cambiar estado de etiqueta de trabajo.			

ID	P 1-6	Fecha	15/09/16	Resultado	Satisfactorio
Finalidad		Refrescar tablero de planificación de trabajo.			
ID	P 1-7	Fecha	15/09/16	Resultado	Satisfactorio
Finalidad		Guardar cambios en tablero de planificación de trabajo.			

Tabla 25 – Pruebas finales de planificación de actividades de trabajo

- **Pruebas de gestión y elaboración de trabajos:**

Estas pruebas están destinadas a comprobar la correcta realización de los procesos ETL, es decir, el correcto mapeo de datos, filtrado de registros y relaciones entre los distintos componentes y demás funcionalidades soportadas. Además, comprobarán la correcta organización de los trabajos en áreas de trabajo y el acceso a esta información.

ID	P 2-1	Fecha	16/09/16	Resultado	Satisfactorio
Finalidad		Acceder al módulo de áreas de trabajo.			
ID	P 2-2	Fecha	16/09/16	Resultado	Satisfactorio
Finalidad		Acceder a un área de trabajo.			
ID	P 2-3	Fecha	16/09/16	Resultado	Satisfactorio
Finalidad		Añadir trabajo a área de trabajo.			
ID	P 2-4	Fecha	16/09/16	Resultado	Satisfactorio
Finalidad		Eliminar trabajo de área de trabajo.			
ID	P 2-5	Fecha	16/09/16	Resultado	Satisfactorio
Finalidad		Buscar trabajo de área de trabajo.			
ID	P 2-6	Fecha	16/09/16	Resultado	Satisfactorio
Finalidad		Buscar versión de trabajo de repositorio.			
ID	P 2-7	Fecha	16/09/16	Resultado	Satisfactorio
Finalidad		Eliminar versión de trabajo de repositorio.			
ID	P 2-8	Fecha	16/09/16	Resultado	Satisfactorio
Finalidad		Elaborar proceso ETL completo en trabajo.			
ID	P 2-9	Fecha	16/09/16	Resultado	Satisfactorio
Finalidad		Subir versión de trabajo a repositorio.			

Tabla 26 – Pruebas finales de gestión y elaboración de trabajos

- **Pruebas de programación y ejecución de trabajos:**

Estas pruebas están destinadas a comprobar el correcto funcionamiento de la programación y ejecución de los trabajos ETL. Comprobando que un trabajo realiza correctamente la labor para la que fue creada y que la ejecución se realiza en el momento indicado en la programación. Además, comprueban que la información que aporta el sistema acerca de la última ejecución de un trabajo sea correcta.

ID	P 3-1	Fecha	17/09/16	Resultado	Satisfactorio
Finalidad		Acceder a módulo de programación de trabajos.			
ID	P 3-2	Fecha	17/09/16	Resultado	Satisfactorio
Finalidad		Programar la ejecución de una versión de un trabajo.			
ID	P 3-3	Fecha	17/09/16	Resultado	Satisfactorio
Finalidad		Eliminar la programación de la ejecución de una versión de un trabajo.			
ID	P 3-4	Fecha	17/09/16	Resultado	Satisfactorio
Finalidad		Ejecutar programación de una versión de un trabajo.			
ID	P 3-5	Fecha	17/09/16	Resultado	Satisfactorio
Finalidad		Consultar información de última ejecución de los trabajos.			

Tabla 27- Pruebas finales de programación y ejecución de trabajos

- **Pruebas de administración:**

Este tipo de pruebas se centran en el área de administración para la gestión de las áreas de trabajo y de los usuarios de la aplicación.

ID	P 4-1	Fecha	17/09/16	Resultado	Satisfactorio
Finalidad		Acceder al área de administración.			
Pasos		1. Loguearse con rol de administrador.			
ID	P 4-2	Fecha	17/09/16	Resultado	Satisfactorio
Finalidad		Visualizar áreas de trabajo.			
ID	P 4-3	Fecha	17/09/16	Resultado	Satisfactorio
Finalidad		Buscar área de trabajo.			
ID	P 4-4	Fecha	17/09/16	Resultado	Satisfactorio
Finalidad		Añadir área de trabajo.			
ID	P 4-5	Fecha	17/09/16	Resultado	Satisfactorio

Finalidad	Eliminar área de trabajo.			
ID	P 4-6	Fecha	17/09/16	Resultado Satisfactorio
Finalidad	Visualizar usuarios de un área de trabajo.			
ID	P 4-7	Fecha	17/09/16	Resultado Satisfactorio
Finalidad	Vincular usuario a área de trabajo.			
ID	P 4-8	Fecha	17/09/16	Resultado Satisfactorio
Finalidad	Desvincular usuario de área de trabajo.			
ID	P 4-9	Fecha	17/09/16	Resultado Satisfactorio
Finalidad	Visualizar usuarios.			
ID	P 4-10	Fecha	17/09/16	Resultado Satisfactorio
Finalidad	Buscar usuario.			
ID	P 4-11	Fecha	17/09/16	Resultado Satisfactorio
Finalidad	Filtrar usuarios.			
ID	P 4-12	Fecha	17/09/16	Resultado Satisfactorio
Finalidad	Añadir usuario.			
ID	P 4-13	Fecha	17/09/16	Resultado Satisfactorio
Finalidad	Modificar usuario.			
ID	P 4-14	Fecha	17/09/16	Resultado Satisfactorio
Finalidad	Añadir área de trabajo a usuario.			
ID	P 4-15	Fecha	17/09/16	Resultado Satisfactorio
Finalidad	Desvincular área de trabajo de usuario.			

Tabla 28 - Pruebas finales de administración

4.2.4.4. Resultado de las pruebas

A tenor de los resultados de la batería de pruebas realizada, se concluye que la aplicación funciona según se espera, teniendo en cuenta que las pruebas de rendimiento no son clave en este proyecto.

5. Evaluación

En este capítulo se presentan los procesos llevados a cabo para determinar si la aplicación realizada funciona según se espera.

Para poder realizar esta evaluación se seleccionarán las distintas funcionalidades de la aplicación en primer lugar y, acto seguido, se procederá a analizarlas.

Se empleará la siguiente estructura de evaluación:

- Planificar actividades de trabajo
- Gestionar y elaborar trabajos
- Programar y ejecutar trabajos
- Administrar áreas de trabajo y usuarios
- Reiniciar clave de acceso y acceder al sistema

En los siguientes puntos se explicará la evaluación que se ha realizado y los resultados derivados de ésta.

5.1. Proceso de evaluación

Esta primera fase de evaluación incluye la elección y el diseño del método de evaluación que se aplica a la herramienta desarrollada y la posterior ejecución de este método.

5.1.1. Diseño de la evaluación

Se ha elaborado un ramillete de escenarios que permitan mostrar cómo funciona la herramienta en manos de un usuario. Estos escenarios deben reflejar los trabajos que se pueden llevar a cabo con la aplicación y comprobar que todos los requisitos del sistema son atendidos.

Se contará con unos cuantos escenarios diferentes. En cada uno de éstos se mostrará el mayor número posible de casos de uso que se obtuvieron en la fase de análisis. Además, hay que mencionar que estos escenarios están emparentados con las pruebas.

Es importante señalar que, debido a la cantidad de funcionalidades que conlleva la aplicación, cada uno de los escenarios que se van a presentar implicará varias tareas que el usuario podría realizar por separado.

Para exponer los escenarios se empleará una tabla a modo de plantilla que contiene la siguiente información:

- **Identificador:** nomenclatura que identifica inequívocamente al escenario. Se define como E-X, donde X es un número que representa el escenario.
- **Nombre:** breve descripción del escenario.
- **Descripción:** descripción detallada del escenario.
- **Tareas a realizar:** descripción guiada de los pasos que componen el escenario.
- **Comprobación:** resultados generados tras la ejecución del escenario.

El modelo de plantilla a seguir, con los campos arriba mencionados, es el siguiente:

Identificador	Nombre
Descripción	
Tareas a realizar	
Comprobación	

Tabla 29 - Formato de tabla de evaluación de escenarios

5.1.2. Escenarios de evaluación

A partir de la plantilla de evaluación de escenarios mostrada en el punto anterior, se inicia la evaluación de los escenarios que tienen como base las principales funcionalidades de la herramienta, es decir, planificación de actividades de trabajo, gestión y elaboración de trabajos, programación y ejecución de trabajos, administración de áreas de trabajo y de usuarios y, por último, reinicio de claves de usuario y acceso al sistema.

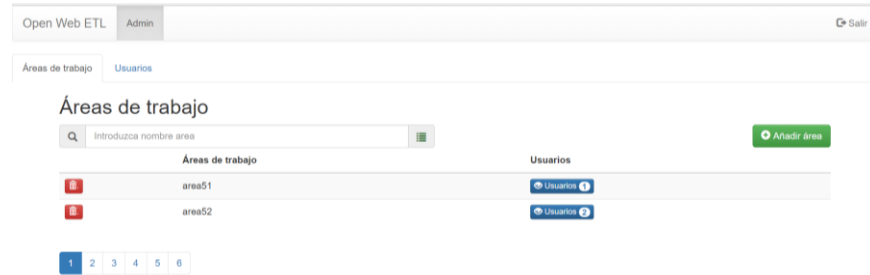
Identificador	E-1	Nombre	Administración de áreas de trabajo y usuarios.
Descripción	<p>El usuario quiere administrar las áreas de trabajo y los usuarios del sistema. En este caso, crea un nuevo área de trabajo, le busca y le asigna dos usuarios. Se da cuenta que uno de los usuarios está mal vinculado y lo elimina. Además, elimina un área de trabajo existente incluyendo sus trabajos y usuarios vinculados. Acto seguido, pasa a gestionar los usuarios. Añade un nuevo usuario con perfil de Usuario, le busca y le vincula el área de trabajo que había creado antes y otra más. Se da cuenta que se ha equivocado y elimina el último área de trabajo que ha vinculado. Seguidamente, filtra los usuarios por rol de Usuario y estado activo. A continuación, cambia el estado de uno de los usuarios para que pase a estar inactivo.</p>		

Tareas a realizar	<ul style="list-style-type: none"> • Seleccionar la pestaña “Áreas de trabajo”. • Hacer clic en el botón “Añadir área” para abrir el panel destinado a añadir un nuevo área de trabajo. • Insertar nombre del nuevo área en el campo etiquetado como “Nombre”. • Hacer clic en el botón “Añadir”. • Seleccionar nombre del área de trabajo añadida en el campo selector descrito como “Introduzca nombre area”. • El usuario comprueba que el área de trabajo se ha creado. • Hacer clic en el botón “Usuarios” asociado al área de trabajo para abrir panel de usuarios. • Seleccionar usuario por el nombre en el campo selector etiquetado como “Nombre de usuario”. • Hacer clic en el botón “Añadir”. • Seleccionar usuario por el nombre en el campo selector etiquetado como “Nombre de usuario”. • Hacer clic en el botón “Añadir”. • El usuario comprueba que se muestran los dos usuarios vinculados. • Hacer clic en el botón con el símbolo de la papelera asociado al segundo usuario para desvincularlo del área de trabajo. • Hacer clic en el botón “Eliminar” del panel de confirmación de borrado. • El usuario comprueba que el usuario eliminado ha desaparecido del listado de usuarios del panel. • Seleccionar nombre del área de trabajo a eliminar en el campo selector descrito como “Introduzca nombre area”. • Hacer clic en el botón con el símbolo de la papelera asociado al área de trabajo seleccionada. • Hacer clic en el botón “Eliminar” del panel de confirmación de borrado. • Seleccionar nombre del área de trabajo a eliminar en el campo selector descrito como “Introduzca nombre area”. • El usuario comprueba que la búsqueda no devuelve el nombre del área de trabajo eliminada. • Seleccionar la pestaña “Usuarios”. • Hacer clic en el botón “Añadir usuario” para abrir el panel destinado a añadir un nuevo usuario. • Insertar nick, nombre, correo y rol del nuevo usuario en los campos etiquetados como “Nick”, “Nombre”, “Correo” y “Roles” (el rol que selecciona el usuario es el de Usuario) respectivamente. • Hacer clic en el botón “Añadir”. • Cerrar mensaje informativo. • Seleccionar nombre del usuario añadido en el campo selector descrito como “Introduzca nombre usuario”. • El usuario comprueba que el usuario se ha creado.
--------------------------	--

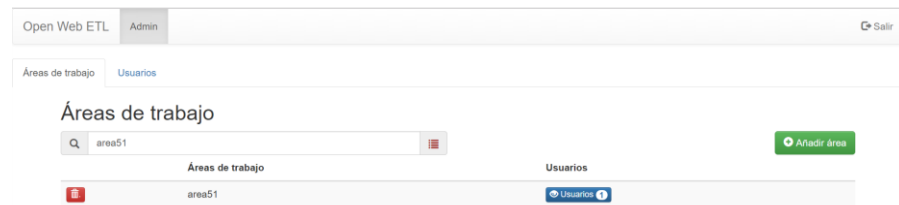
	<ul style="list-style-type: none">• Hacer clic en el botón “Áreas” asociado al usuario para abrir panel de áreas de trabajo.• Seleccionar la nueva área creada anteriormente por el nombre en el campo selector etiquetado como “Área de trabajo”.• Hacer clic en el botón “Añadir”.• Seleccionar otro área por el nombre en el campo selector etiquetado como “Área de trabajo”.• Hacer clic en el botón “Añadir”.• El usuario comprueba que se muestran las dos áreas vinculadas.• Hacer clic en el botón con el símbolo de la papelera asociado al segundo área de trabajo para desvincularlo del usuario.• Hacer clic en el botón “Eliminar” del panel de confirmación de borrado.• El usuario comprueba que el área ha desaparecido del listado de áreas de trabajo del panel.• Cerrar panel de áreas de trabajo vinculadas.• Hacer clic en el botón desplegable “Filtro”.• Seleccionar la opción “Usuarios”.• Seleccionar la opción “Activos”.• El usuario comprueba que en la tabla de usuario sólo aparecen los usuarios activos con rol Usuario.• Seleccionar nombre del usuario a modificar en el campo selector descrito como “Introduzca nombre usuario”.• Hacer clic en el botón con el símbolo de edición asociado al usuario seleccionado para abrir el panel de edición.• Cambiar el estado a “No activo”.• Hacer clic en el botón “Editar”.• Cerrar mensaje informativo.• Seleccionar nombre del usuario modificado en el campo selector descrito como “Introduzca nombre usuario”.• El usuario comprueba que el estado del usuario modificado está marcado con un aspa de color rojo.
--	---

Comprobación

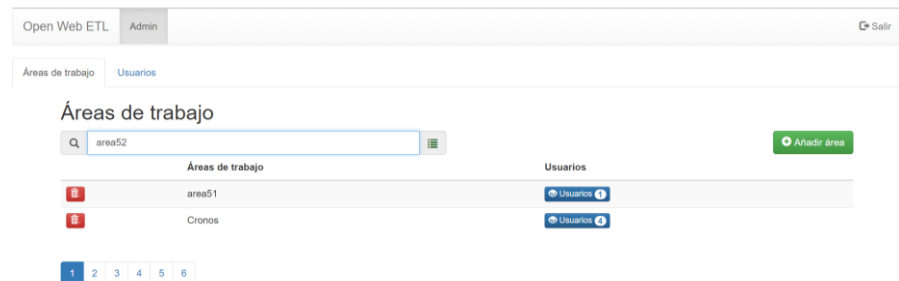
El usuario comprueba que se encuentra en la vista de gestión de áreas de trabajo visualizando la cabecera “Áreas de trabajo”:



El usuario comprueba que el área de trabajo que ha incluido se ha añadido correctamente buscando el nombre de la misma en el buscador de áreas de trabajo y viendo que aparece:

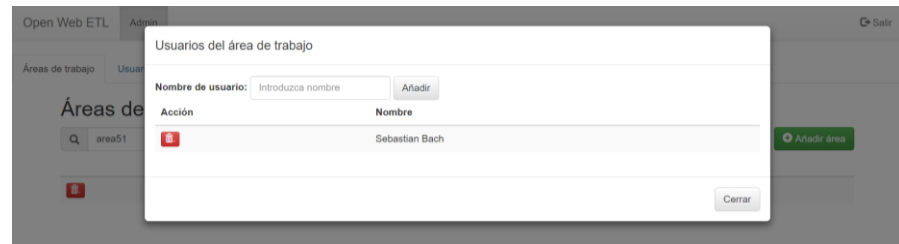


El usuario comprueba que el área de trabajo que ha eliminado se ha borrado correctamente buscando el nombre de la misma en el buscador de áreas de trabajo y viendo que no aparece:

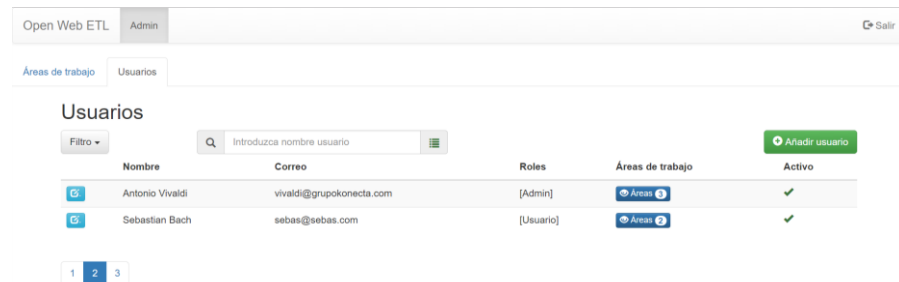


El usuario comprueba que realmente se ha vinculado el usuario al área de trabajo según ha indicado, buscando el área de trabajo en cuestión y comprobando que el número de usuarios que indica el botón “Usuarios” es correcto y verificando en el panel de usuario asociado al área de trabajo que aparece el usuario en el listado de usuarios. Del mismo modo, en este

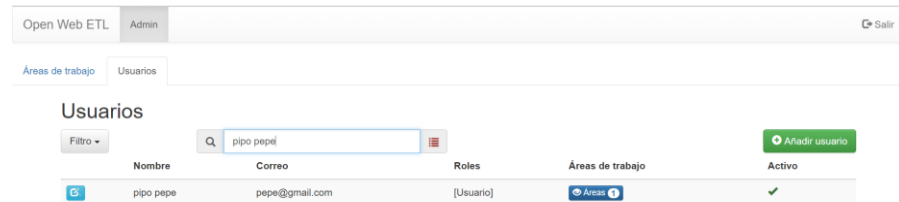
panel comprueba que el usuario que había desvinculado no aparece en la lista de usuarios:



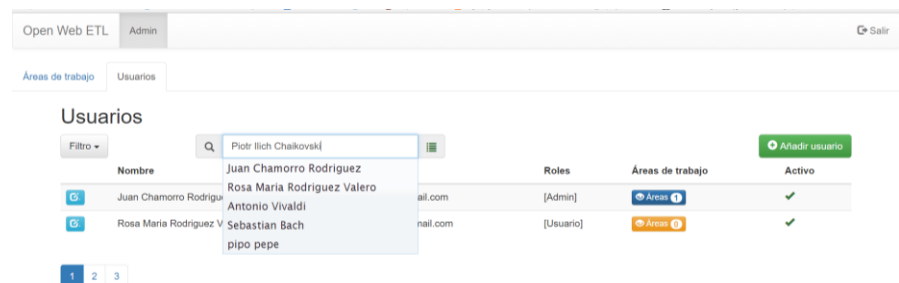
El usuario comprueba que se encuentra en la vista de gestión de usuarios visualizando la cabecera “Usuarios”:



El usuario comprueba que el usuario que ha incluido se ha añadido correctamente buscando el nombre del mismo en el buscador de usuarios y viendo que aparece:

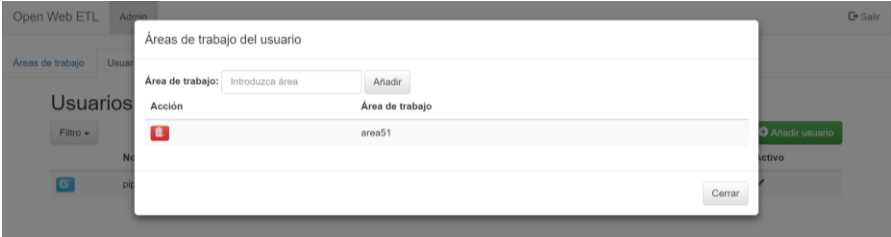


El usuario comprueba que el usuario que ha eliminado se ha borrado correctamente buscando el nombre del mismo en el buscador de usuarios y viendo que no aparece:



El usuario comprueba que realmente se ha vinculado el área de trabajo al usuario según ha indicado, buscando el usuario en cuestión y comprobando que el número de áreas que indica el botón “Áreas” es correcto y verificando en el panel de áreas de trabajo asociado al usuario que aparece el área en el listado de áreas de trabajo. Del mismo modo, en

este panel comprueba que el área que había desvinculado no aparece en la lista de áreas de trabajo:



El usuario comprueba que el usuario que ha desactivado lo está realmente buscando el nombre del mismo en el buscador de usuarios y comprobando que en su columna de estado aparece un aspa de color rojo:

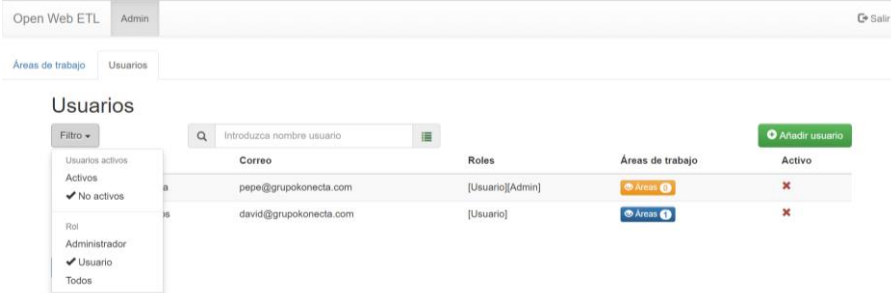


Tabla 30 - Escenario E-1

Identificador	E-2	Nombre	Planificación de actividades de trabajo.
Descripción	El usuario quiere planificar las tareas que componen su trabajo para poder organizarse. Con este fin comprueba las actividades y el estado de las mismas que refleja su tablero de planificación. Acto seguido, crea una actividad y la mantiene en su estado inicial. A continuación, elimina una actividad que está en el estado de 'Hecho' y cambia una actividad de estado 'Pendiente' a 'En curso'. Finalmente, guarda los cambios de su tablero.		

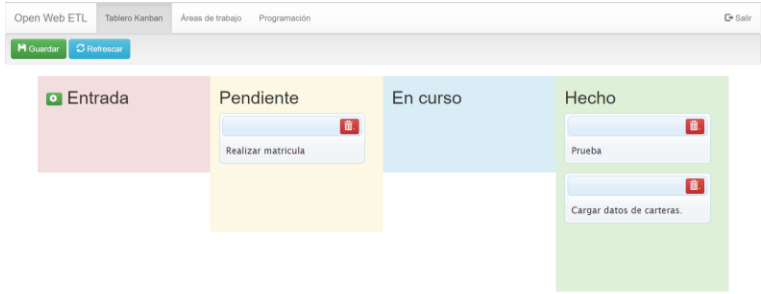

Tareas a realizar	<ul style="list-style-type: none"> • Seleccionar la pestaña “Tablero Kanban”. • Hacer clic en el botón con el símbolo más junto a la cabecera “Entrada” para abrir el panel para añadir nueva actividad. • Introducir descripción de la actividad en el campo “Contenido” del panel y pulsar el botón “Añadir”. • Hacer clic en el botón “Guardar”. • El usuario comprueba que la nueva etiqueta aparece en la columna de estado ‘Entrada’. • Hacer clic en el botón con el símbolo de una papelera asociado a una actividad de la columna de estado ‘Hecho’ para eliminarla. • El usuario comprueba que la actividad ha desaparecido de esa columna de estado. • Hacer clic sobre la cabecera de una actividad de la columna de estado ‘Pendiente’, sin soltar el ratón desplazar la etiqueta a la columna de estado ‘En curso’ y soltar el ratón (arrastrar y soltar). • El usuario comprueba que la actividad ha pasado de la columna ‘Pendiente’ a la columna ‘En curso’. • Hacer clic en el botón “Guardar”. • El usuario comprueba que los cambios en su tablero de planificación se han guardado con éxito cuando visualiza la ventana modal informativa en la que se confirma esta operación.
Comprobación	<p>El usuario comprueba que tras pulsar el botón ‘Refrescar’ su tablero de planificación de actividades de trabajo refleja los cambios que había realizado sobre éste y que había guardado.</p> <p>Estado inicial del tablero de planificación del usuario antes de realizar los cambios:</p>  <p>Estado del tablero de planificación del usuario después de haber realizado los cambios:</p> 

Tabla 31 - Escenario E-2

Identificador	E-3	Nombre	Gestión de trabajos.
Descripción	El usuario quiere gestionar los trabajos de un área de trabajo a la que está vinculada. Con este fin, accede al área de trabajo y crea un trabajo. Acto seguido, abre un trabajo existente y busca una versión de repositorio y la elimina. Por último, elimina el trabajo que tiene abierto.		
Tareas a realizar	<ul style="list-style-type: none"> • Seleccionar la pestaña “Áreas de trabajo”. • Seleccionar área de trabajo del campo selector etiquetado como “Área seleccionada”. • Hacer clic en el botón “Seleccionar”. • El usuario comprueba que está en el área de trabajo que ha seleccionado. • Hacer clic en el botón “Añadir trabajo” en la vista del área de trabajo previamente seleccionada para abrir panel para añadir un nuevo trabajo. • Indicar nombre y descripción de trabajo en los campos habilitados para ello. • Hacer clic en botón “Añadir”. • Cerrar mensaje informativo. • Seleccionar trabajo del campo selector etiquetado como “Seleccione un trabajo de la lista”. • El usuario comprueba que aparece el nombre del nuevo trabajo añadido, pero selecciona otro que existía previamente. • Seleccionar versión del trabajo existente del campo selector etiquetado como “Seleccione una versión” en la vista del trabajo cargada. • Hacer clic en el botón “Eliminar versión repositorio”. • Hacer clic en el botón “Eliminar” del panel de confirmación. • Cerrar mensaje informativo. • El usuario comprueba que la versión del trabajo ha desaparecido de la vista y no aparece en el listado de versiones. • Hacer clic en el botón “Eliminar trabajo”. • Hacer clic en el botón “Eliminar” del panel de confirmación. • Cerrar mensaje informativo. • El usuario comprueba que la información del trabajo ha desaparecido de la vista y no aparece en el listado de trabajos. 		

Comprobación	<p>El usuario comprueba que se encuentra en el área de trabajo que ha seleccionado viendo que el nombre de la cabecera de la vista se corresponde con el de dicho área:</p> <div data-bbox="451 360 1342 495"><p>Open Web ETL Tablero Kanban Áreas de trabajo Programación Salir</p><p>Área de trabajo: Cronos Añadir trabajo</p><p>Seleccione un trabajo de la lista:</p><input type="text"/></div> <p>El usuario comprueba que se ha creado el trabajo añadido al área de trabajo en el que se encuentra consultando su nombre en el buscador de trabajos y cargando sus datos en la vista:</p> <div data-bbox="451 685 1342 1055"><p>Open Web ETL Tablero Kanban Áreas de trabajo Programación Salir</p><p>Área de trabajo: Cronos Añadir trabajo</p><p>Seleccione un trabajo de la lista:</p><input type="text"/></div> <div data-bbox="501 833 1294 1055"><p>Trabajo: CargarCarteras Eliminar trabajo</p><p>Insertar registros en BBDD de Carteras.</p><table><thead><tr><th>Copia de usuario</th><th>Repositorio</th></tr></thead><tbody><tr><td>No existe copia.</td><td>Repositorio vacío.</td></tr><tr><td>Abrir nuevo trabajo usuario</td><td>Eliminar versión repositorio</td></tr><tr><td></td><td>Abrir trabajo repositorio</td></tr></tbody></table></div> <p>El usuario comprueba que se ha eliminado la versión de repositorio del trabajo abierto consultando el selector de versiones y viendo que no se encuentra en éste (en este caso se ha eliminado “Versión 1”):</p> <div data-bbox="451 1227 1342 1599"><p>Open Web ETL Tablero Kanban Áreas de trabajo Programación Salir</p><p>Área de trabajo: Cronos Añadir trabajo</p><p>Seleccione un trabajo de la lista:</p><input type="text"/></div> <div data-bbox="501 1375 1294 1599"><p>Trabajo: CargarGeos Eliminar trabajo</p><p>Carga de Geos en BBDD.</p><table><thead><tr><th>Copia de usuario</th><th>Repositorio</th></tr></thead><tbody><tr><td>Abrir trabajo usuario</td><td>Seleccione una versión:</td></tr><tr><td></td><td><div>Versión 2</div></td></tr><tr><td></td><td>Abrir trabajo repositorio</td></tr></tbody></table></div> <p>El usuario comprueba que ya no existe el trabajo que ha eliminado viendo que los datos del mismo han desaparecido de la vista y que su nombre no aparece en el buscador de trabajos:</p> <div data-bbox="451 1771 1342 1906"><p>Open Web ETL Tablero Kanban Áreas de trabajo Programación Salir</p><p>Área de trabajo: Cronos Añadir trabajo</p><p>Seleccione un trabajo de la lista:</p><input type="text"/></div>	Copia de usuario	Repositorio	No existe copia.	Repositorio vacío.	Abrir nuevo trabajo usuario	Eliminar versión repositorio		Abrir trabajo repositorio	Copia de usuario	Repositorio	Abrir trabajo usuario	Seleccione una versión:		<div>Versión 2</div>		Abrir trabajo repositorio
Copia de usuario	Repositorio																
No existe copia.	Repositorio vacío.																
Abrir nuevo trabajo usuario	Eliminar versión repositorio																
	Abrir trabajo repositorio																
Copia de usuario	Repositorio																
Abrir trabajo usuario	Seleccione una versión:																
	<div>Versión 2</div>																
	Abrir trabajo repositorio																

Tabla 32 - Escenario E-3

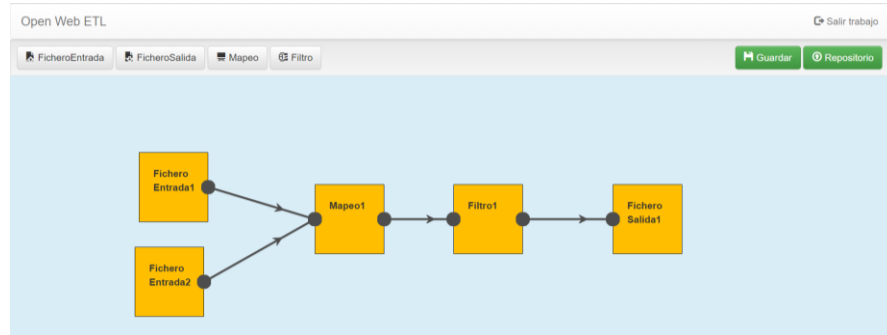
Identificador	E-4	Nombre	Elaboración de trabajo ETL.
Descripción	<p>El usuario quiere realizar un proceso ETL en el trabajo al que ha accedido. Con este fin, extrae los datos de dos ficheros de tipo CSV de entrada con las columnas separados por ';' y los mapea por medio de los dos primeros campos de cada fichero de tipo 'String' mediante una relación de <i>inner join</i>. Además, elimina el segundo campo del segundo fichero que actúa como secundario. Acto seguido, filtra las filas en las que el valor del primer campo que actúa como principal sea igual a 'OK' descartando las demás. A continuación, conecta el flujo de datos con un fichero de salida de tipo CSV. Por último, sube una versión del trabajo al repositorio y guarda una copia de usuario.</p>		

<p>Tareas realizar</p>	<p>a</p> <ul style="list-style-type: none"> • Seleccionar la pestaña “Áreas de trabajo”. • Seleccionar área de trabajo del campo selector etiquetado como “Área seleccionada”. • Hacer clic en el botón “Seleccionar”. • Seleccionar trabajo del campo selector etiquetado como “Seleccione un trabajo de la lista”. • Seleccionar versión del trabajo existente del campo selector etiquetado como “Seleccione una versión” en la vista del trabajo cargada. • Hacer clic en el botón “Abrir trabajo repositorio”. • El usuario comprueba que ha accedido al área de elaboración de procesos ETL del trabajo seleccionado. • Hacer clic en el botón “FicheroEntrada” del panel de componentes. • Hacer clic en el área de trabajo. • Hacer doble clic sobre el componente “FicheroEntrada” para abrir su panel de gestión. • El usuario comprueba que el componente se ha creado en la vista. • Seleccionar pestaña “Mapeo campos” del panel. • Introducir nombre del campo y seleccionar tipo de campo ‘String’. • Hacer clic en botón “Añadir”. • Seleccionar pestaña “Fichero”. • Introducir ruta completa del fichero CSV y como campo separador de columna ‘;’ en los campos habilitados. • Hacer clic en botón “Aplicar”. • Hacer clic en botón “Cerrar” para cerrar panel de “FicheroEntrada”. • Hacer clic en el botón “FicheroEntrada” del panel de componentes. • Hacer clic en el área de trabajo. • El usuario comprueba que el componente se ha creado en la vista. • Hacer doble clic sobre el componente “FicheroEntrada” para abrir su panel de gestión. • Seleccionar pestaña “Mapeo campos” del panel. • Introducir nombre del campo y seleccionar tipo de campo ‘String’. • Hacer clic en botón “Añadir”. • Introducir el nombre de otro campo y seleccionar tipo de campo ‘Boolean’. • Hacer clic en botón “Añadir”. • Seleccionar pestaña “Fichero”. • Introducir ruta completa del fichero CSV y campo separador de columna en los campos habilitados. • Hacer clic en botón “Aplicar”. • Hacer clic en botón “Cerrar” para cerrar panel de “FicheroEntrada”. • Hacer clic en el botón “Mapeo” del panel de componentes. • Hacer clic en el área de trabajo. • El usuario comprueba que el componente se ha creado en la vista. • Hacer clic en el botón “Filtro” del panel de componentes. • Hacer clic en el área de trabajo.
-------------------------------	---

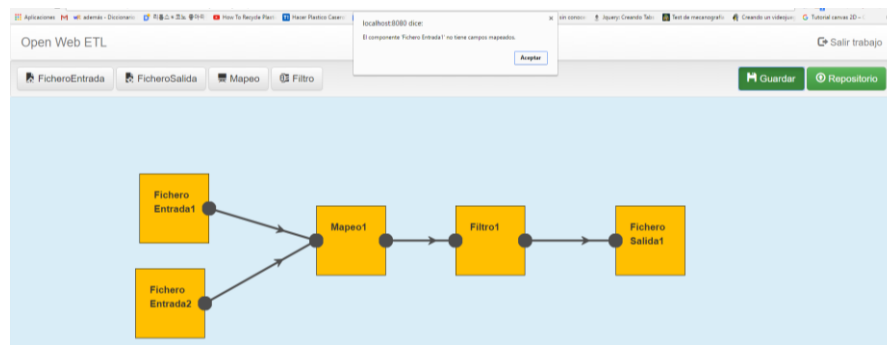
- El usuario comprueba que el componente se ha creado en la vista.
- Hacer clic en el botón “FicheroSalida” del panel de componentes.
- Hacer clic en el área de trabajo.
- El usuario comprueba que el componente se ha creado en la vista.
- Hacer doble clic sobre el primer componente “FicheroEntrada” para abrir su panel de gestión.
- En la pestaña “Conector” del panel seleccionar el nombre del componente “Mapeo” del campo selector de conectores de salida.
- Hacer clic en el botón “Aplicar”.
- El usuario comprueba que se ha creado una flecha que conecta el primer componente “FicheroEntrada” con el componente “Mapeo”.
- Hacer doble clic sobre el segundo componente “FicheroEntrada” para abrir su panel de gestión.
- En la pestaña “Conector” del panel seleccionar el nombre del componente “Mapeo” del campo selector de conectores de salida.
- Hacer clic en el botón “Aplicar”.
- El usuario comprueba que se ha creado una flecha que conecta el segundo componente “FicheroEntrada” con el componente “Mapeo”.
- Hacer doble clic sobre el componente “Mapeo” para abrir su panel de gestión.
- Seleccionar la pestaña “Mapeo campos” del panel.
- Hacer clic en el botón con el símbolo de una papelera asociado al tercer registro que se corresponde con el segundo campo mapeado en el segundo componente “FicheroEntrada”.
- Seleccionar la pestaña “Relaciones”.
- Seleccionar en el primer selector con los campos del primer componente “FicheroEntrada” el primer campo.
- Seleccionar en el segundo selector con los campos del segundo componente “FicheroEntrada” el primer campo.
- Seleccionar el tipo de relación “Inner join”.
- Hacer clic en el botón con el símbolo de confirmación.
- Seleccionar la pestaña “Conector”.
- Seleccionar el nombre del componente “Filtro” del campo selector de conectores de salida.
- Hacer clic en el botón “Aplicar”.
- El usuario comprueba que se ha creado una flecha que conecta el componente “Mapeo” con el componente “Filtro”.
- Hacer doble clic sobre el componente “Filtro” para abrir su panel de gestión.
- Seleccionar la pestaña “Filtros”.
- Seleccionar el campo correspondiente al primer campo mapeado en el primer componente “FicheroEntrada” en el campo selector de campos.
- Seleccionar que el campo va a ser igual al valor ‘OK’.
- Hacer clic en el botón con el símbolo de confirmación.
- Seleccionar la pestaña “Conector”

	<ul style="list-style-type: none"> • Seleccionar el nombre del componente “FicheroSalida” del campo selector de conectores de salida. • Hacer clic en el botón “Aplicar”. • El usuario comprueba que se ha creado una flecha que conecta el componente “Filtro” con el componente “FicheroSalida”. • Hacer doble clic sobre el componente “FicheroSalida” para abrir su panel de gestión. • Seleccionar la pestaña “Fichero”. • Indicar la ruta del fichero CSV de salida en el campo etiquetado como “Ruta fichero”. • Hacer clic en el botón “Aplicar”. • Hacer clic en el botón “Guardar”. • El usuario comprueba que no se ha abierto una ventana modal mostrando errores. • Cerrar mensaje informativo. • Hacer clic en el botón “Repositorio” para abrir panel de repositorio del trabajo. • Seleccionar número de versión del campo selector etiquetado como “Versión”. • Escribir la descripción de la versión en el campo etiquetado como “Descripción”. • Hacer clic en el botón “Subir”. • Hacer clic en el botón “Confirmar” del panel de confirmación. • Cerrar mensaje informativo. • El usuario vuelve a abrir el panel de repositorio de trabajo y comprueba que aparece la versión subida. • Pulsar botón “Salir trabajo” situado en la cabecera.
Comprobación	El usuario comprueba que el proceso ETL que ha implementado se corresponde con el que se muestra en la vista. Es decir, comprueba que los

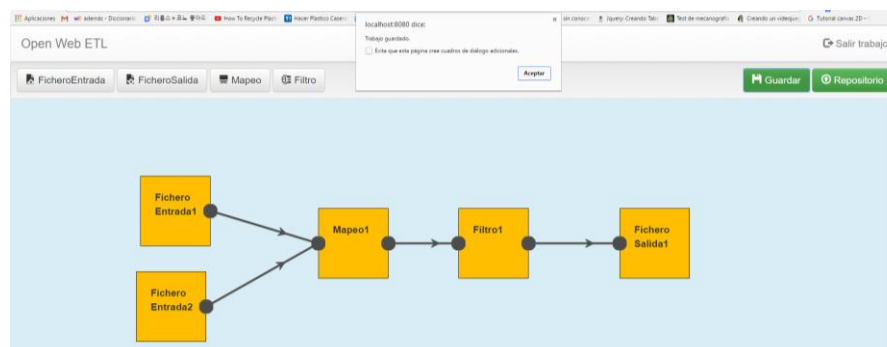
componentes son los correctos y que las conexiones entre ellos son las que deberían:



El usuario comprueba que el proceso ETL que ha construido es correcto si al pulsar el botón “Guardar” no aparece un mensaje modal de errores como el siguiente:



El usuario comprueba que se ha guardado una copia de usuario del trabajo visualizando el mensaje informativo que confirma la operación tras haber pulsado el botón “Guardar”.



El usuario comprueba que se ha guardado la versión de repositorio que ha subido abriendo el panel del repositorio y comprobando que aparece una fila con los datos de esta versión:

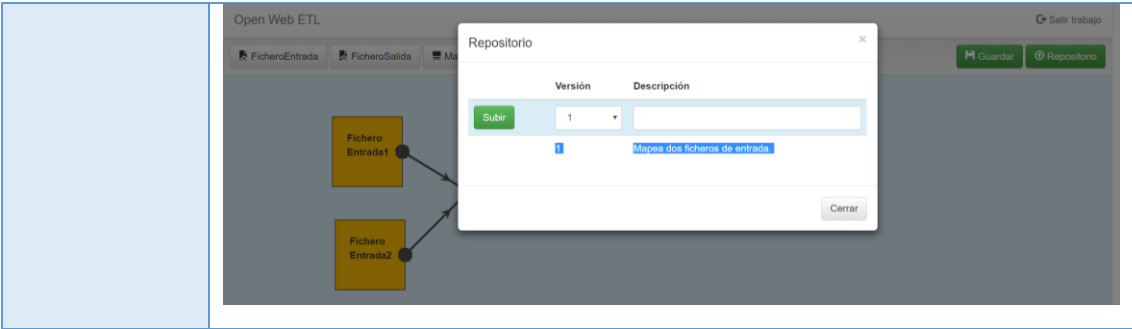


Tabla 33 - Escenario E-4

Identificador	E-5	Nombre	Programación y ejecución de trabajos.
Descripción	<p>El usuario quiere programar un trabajo ETL que está dentro de un área de trabajo a la que tiene acceso y consultar el resultado de la ejecución. Con este fin accede a la vista de planificación de trabajos y selecciona el nombre del trabajo, la versión del mismo y el horario de programación. Acto seguido, pulsa el botón correspondiente para añadir la nueva planificación y observa que se ha añadido en la lista de trabajos programados. Espera hasta después de la hora en la que programó el trabajo y refresca la vista de planificación de trabajos. A continuación, comprueba que se haya ejecutado el trabajo visualizando su información de fecha, estado y descripción de la última ejecución. Seguidamente, accede al directorio donde el proceso debería haber generado el fichero CSV de salida y lo abre para verificar que el resultado es correcto (el directorio pertenece a una unidad montada tanto en el equipo del cliente como en el servidor donde está desplegada la aplicación). Por último, elimina la última programación ejecutada.</p>		
Tareas realizar	a	<ul style="list-style-type: none">• Seleccionar la pestaña “Programación”.• Seleccionar trabajo del campo selector “Nombre trabajo”.• Seleccionar versión de trabajo del campo selector “Versión”.• Escribir el horario de programación en el campo de texto “Programación”.• Hacer clic en botón con el símbolo de confirmación.• Cerrar el mensaje de confirmación de alta.• El usuario comprueba que se ha añadido una nueva fila con la nueva programación a la tabla de programaciones y espera a que pase la hora de ejecución.• El usuario consulta los campos “Última ejecución”, “Estado” y “Descripción” del trabajo ejecutado.• El usuario accede al directorio donde el proceso debería haber generado el fichero CSV de salida y comprueba que existe y es válido.• Hacer clic en el botón con el símbolo de la papelera asociado a la programación ejecutada.• Cerrar el mensaje de confirmación de borrado.• El usuario comprueba que la fila de programación ha desaparecido de la tabla de programaciones.	

Comprobación

n

El usuario comprueba que la programación para la versión del trabajo añadida aparece en la tabla de programaciones de la vista.

Vista antes de añadir la nueva programación:

Open Web ETL

Tablero Kanban

Áreas de trabajo

Programación

Sair

Programación de procesos

Nombre trabajo	Versión	Programación	Última ejecución	Estado	Descripción
 CargarGeos	1				

Vista después de añadir la nueva programación:

Open Web ETL

Tablero Kanban

Áreas de trabajo

Programación

Sair

Programación de procesos

Nombre trabajo	Versión	Programación	Última ejecución	Estado	Descripción
 CargarGeos	2	0 0 21 * * ?			
 CargarGeos		0 0 21 * * ?	nunca		

El usuario espera hasta después de la hora en la que programó el trabajo y refresca la vista “Programación de procesos”. A continuación, consulta los campos “Última ejecución”, “Estado” y “Descripción” de este trabajo y verifica que la ejecución se ha realizado:

Open Web ETL

Tablero Kanban

Áreas de trabajo

Programación

Sair

Programación de procesos

Nombre trabajo	Versión	Programación	Última ejecución	Estado	Descripción
 CargarGeos	2	0 0 21 * * ?			
 CargarGeos		0 0 21 * * ?	2016-09-18T19:03:00.653Z	OK	Trabajo ejecutado correctamente.

El usuario accede al directorio donde el proceso debería haber generado el fichero CSV de salida y comprueba que existe y es válido (“ficheroResultado.csv”):

```

root@juan-pc:~/etc# cd /home/juan
root@juan-pc:/home/juan# ls -lh
total 156
drwxr-xr-x 2 juan juan 4,0K Jun 15 17:33 Descargas
drwxr-xr-x 4 juan juan 4,0K Sep 18 2015 Documentos
drwxr-xr-x 2 juan juan 4,0K Dic 8 2014 Escritorio
drwxr-xr-x 2 juan juan 4,0K Sep 18 2015 Idiomas
drwxr-xr-x 2 juan juan 4,0K Dic 8 2014 Música
drwxr-xr-x 2 juan juan 4,0K Dic 8 2014 Plantillas
drwxr-xr-x 2 juan juan 4,0K Dic 8 2014 Público
drwxr-xr-x 2 juan juan 4,0K Dic 8 2014 Videos
root@juan-pc:/home/juan#

```

El usuario comprueba que la fila de programación eliminada ha desaparecido de la tabla de programaciones:

Open Web ETL

Tablero Kanban

Áreas de trabajo

Programación

Sair

Programación de procesos

Nombre trabajo	Versión	Programación	Última ejecución	Estado	Descripción
 CargarGeos	2				

Tabla 34 - Escenario E-5

Identificador	E-6	Nombre	Reinicio de clave de usuario y acceso al sistema.
Descripción	El usuario quiere <i>loguearse</i> para poder acceder al área restringida de la aplicación. Con este fin, introduce su <i>nick</i> y clave de acceso en la pantalla de acceso y pulsa el botón para poder acceder, pero se le informa de que		

	<p>la contraseña no es válida. Por este motivo, accede a la pantalla de reinicio de clave e indica su correo. Acto seguido, mira el correo emitido por la aplicación y accede al enlace que le indica. En la pantalla que abre este enlace indica su nueva clave de usuario y accede a la pantalla de acceso para volver a introducir su <i>nick</i> y su nueva clave.</p>
Tareas a realizar	<ul style="list-style-type: none"> • En la página de acceso, introducir el nick y la clave de usuario en los campos correspondientes. • Hacer clic en el botón “Acceder”. • El usuario comprueba que los datos de acceso no son correctos por el panel informativo que se le muestra al respecto. • Hacer clic en el enlace “Reinicio de clave de usuario”. • Introducir correo en el campo etiquetado como “Correo”. • Hacer clic en el botón “Enviar solicitud”. • Comprueba que ha recibido un correo con el asunto “[Open Web ETL] Reinicio de clave”. • Hacer clic en el enlace del correo recibido con el asunto “[Open Web ETL] Reinicio de clave”. • En la página abierta desde el enlace, introducir la nueva clave en el campo etiquetado como “Clave”. • Hacer clic en el botón “Cambiar clave”. • Comprueba que se ha realizado la operación cuando visualizar el mensaje “Cambio de clave correcto”. • Hacer clic en el enlace de la cabecera “Acceder”. • En la página de acceso a la que se ha llegado introducir el nick y la clave de usuario en los campos correspondientes. • Hacer clic en el botón “Acceder”. • El usuario comprueba que se encuentra dentro de la aplicación y que no se ha quedado en la página de acceso.


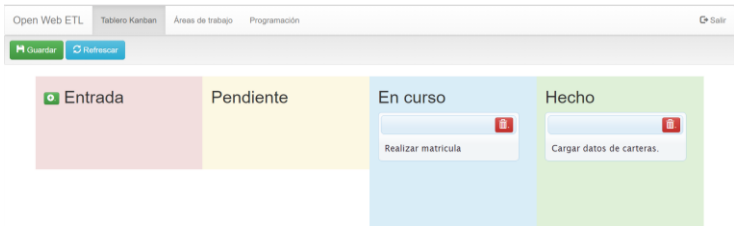
Comprobación	<p>El usuario no puede acceder al contenido de la aplicación cuando introduce mal los datos de acceso en la pantalla de acceso:</p> <div data-bbox="788 353 1034 611"></div> <p>En cambio, cuando introduce los datos de acceso correctos en la pantalla de acceso (con su nueva clave) puede acceder al contenido de la aplicación:</p> <div data-bbox="475 770 1211 994"></div>
---------------------	---

Tabla 35 - Escenario E-6

5.2. Análisis de resultados

En este punto se estudian los resultados que se han generado al ejecutar los escenarios de evaluación planteados en el punto previo. Con este fin, se empleará una matriz de trazabilidad donde se vincula cada requisito planteado en el apartado de requisitos (especificados en el anexo “[Anexo I. Requisitos](#)”) con cada uno de los escenarios de evaluación indicados en el punto previo.

Mediante esta matriz se podrá comprobar de una forma muy visual y simple que se han tenido en cuenta todos los requisitos indicados en los escenarios planteados. Para alcanzar esta meta, es necesario que cada requisito esté vinculado, como mínimo, con un escenario de evaluación.

No pueden existir requisitos sin vínculo. Si se diera este caso, sería necesario incorporar nuevos escenarios de evaluación. Por el contrario, sí es factible que un requisito quede reflejado en más de un escenario, puesto que podría tratarse de una parte fundamental del sistema.

ESCENARIOS REQUISITOS	E-1	E-2	E-3	E-4	E-5	E-6
RRG-01						x
RRG-02						x
RRG-03						x
RFAA-04	x					
RRAA-05	x					
RRAA-06	x					
RRAA-07	x					

ESCENARIOS REQUISITOS	E-1	E-2	E-3	E-4	E-5	E-6
RRAA-08	x					
RRAA-09	x					
RRAA-10	x					
RFAA-11	x					
RRAA-12	x					
RFAA-13	x					
RRAA-14	x					
RRAA-15	x					
RFAA-16	x					
RRAA-17	x					
RRAA-18	x					
RFAA-19	x					
RRAA-20	x					
RRAA-21	x					
RRAA-22	x					
RRAA-23	x					
RFAA-24	x					
RRAA-25	x					
RFAA-26	x					
RRAA-27	x					
RRAA-28	x					
RFAU-29	x					
RRAU-30	x					
RRAU-31	x					
RRAU-32	x					
RRAU-33	x					
RRAU-34	x					
RRAU-35	x					
RRAU-36	x					
RFAU-37	x					
RRAU-38	x					
RRAU-39	x					
RRAU-40	x					
RRAU-41	x					
RFAU-42	x					
RRAU-43	x					
RRAU-44	x					
RFAU-45	x					
RRAU-46	x					
RFAU-47	x					
RRAU-48	x					
RRAU-49	x					
RRAU-50	x					
RRAU-51	x					
RFAU-52	x					
RRAU-53	x					
RFAU-54	x					
RRAU-55	x					
RRAU-56	x					

ESCENARIOS REQUISITOS	E-1	E-2	E-3	E-4	E-5	E-6
RFTP-57		x				
RRTP-58		x				
RRTP-59		x				
RRTP-60		x				
RFTP-61		x				
RRTP-62		x				
RFTP-63		x				
RRTP-64		x				
RFTP-65		x				
RRTP-66		x				
RFTP-67		x				
RRTP-68		x				
RFTP-69		x				
RRTP-70		x				
RFTE-71					x	
RRTE-72					x	
RFTE-73					x	
RRTE-74					x	
RRTE-75					x	
RFTE-76					x	
RRTE-77					x	
RFTE-78					x	
RRTE-79					x	
RRTE-80					x	
RFTG-84			x	x		
RRTG-85			x	x		
RRTG-86			x	x		
RRTG-87			x	x		
RFTG-88			x	x		
RFTG-89			x	x		
RRTG-90			x	x		
RFTG-91			x			
RRTG-92			x			
RRTG-93			x			
RRTG-94			x			
RRTG-95			x			
RFTG-96			x	x		
RRTG-97			x	x		
RFTG-98			x			
RRTG-99			x			
RRTG-100			x			
RFTG-101			x	x		
RRTG-102			x	x		
RFTG-103			x	x		
RRTG-104			x	x		
RFTG-105			x			
RRTG-106			x			
RFTG-107				x		
RRTG-108				x		

ESCENARIOS REQUISITOS	E-1	E-2	E-3	E-4	E-5	E-6
RFTG-109				x		
RRTG-110				x		
RFTG-111				x		
RRTG-112				x		
RRTG-113				x		
RRTG-114				x		
RRTG-115				x		
RRTG-116				x		
RRTG-117				x		
RRTG-118				x		
RRTG-119				x		
RRTG-120				x		
RRTG-121				x		
RRTG-122				x		
RRTG-123				x		
RRTG-124				x		
RRTG-125				x		
RRTG-126				x		
RRTG-127				x		
RRTG-128				x		
RRTG-129				x		
RRTG-130				x		
RRTG-131				x		
RRTG-132				x		
RRTG-133				x		
RRTG-134				x		
RRTG-135				x		
RRTG-136				x		
RRTG-137				x		
RRTG-138				x		
RRTG-139				x		
RRTG-140				x		
RRTG-141				x		
RRTG-142				x		
RRTG-143				x		
RRTG-144				x		
RRTG-145				x		
RRTG-146				x		
RRTG-147				x		
RRTG-148				x		
RRTG-149				x		

Tabla 36 - Matriz de trazabilidad de requisitos vs escenarios de evaluación

6. Conclusión

La presente memoria finaliza con este último capítulo que recapitula los puntos fundamentales del trabajo haciendo una reflexión ordenada sobre los aspectos del tema tratado. Con ese fin, se presentan los objetivos cumplidos, los problemas que han surgido y los conocimientos nuevos que no se tenían antes de la realización del trabajo. Además, se proponen unas líneas de trabajo a seguir en el futuro para continuar enriqueciendo la aplicación gestada en este proyecto.

6.1. Conclusión

Big Data es un término que forma parte del vocabulario empleado dentro de las empresas de hoy en día. Según un gran número de estudios de investigación y de mercado independientes las cantidades de datos se multiplican por dos cada año. Además, un gran número de compañías almacenan los datos históricos de dos o más años. Big Data se encarga de explotar esos volúmenes de información y de nuevos tipos de datos con nuevas clases de herramientas. Muchas de las aplicaciones que se utilizaban para hacer frente a una cantidad de información relativamente grande no pueden sostenerse hoy en día con Big Data. Esa nueva generación de herramientas posee el potencial para transformar la forma en que funciona el negocio de las organizaciones.

Dentro de esta nueva generación de herramientas de Big Data se encuentran las denominadas ETL, software que se ocupa de la integración de Big Data, en particular, de la extracción de datos tanto homogénea como heterogénea, transformar los datos para su almacenamiento en el formato adecuado o estructura para las consultas y análisis y, además, cargar estos datos en base de datos o en el sistema de destino que corresponda.

Las herramientas ETL, de una u otra manera, han existido desde hace más de dos décadas. De hecho, los inicios de estas herramientas se sitúan en la época en la que empezaron a migrarse datos del ordenador central, cuando las personas movían los datos de una aplicación a otra. Por consiguiente, constituye una de las tecnologías de integración de datos más maduras. Sin embargo, a pesar de contar con esa madurez, las herramientas ETL actuales siguen presentando algunas carencias que han sido detectadas durante el estudio llevado a cabo en este trabajo.

Las herramientas ETL existentes no tienen en cuenta el grupo de trabajo que hace uso de las mismas. A este respecto, cada vez es más común encontrar equipos especializados en producir procesos ETL dentro de los departamentos de desarrollo de las empresas y, como en cualquier otro equipo de trabajo, nace el problema de cómo optimizar las tareas y el de cómo acotar responsabilidades. Para poder solucionar el primer problema, es decir, ayudar a gestionar mejor el trabajo, un usuario tendría que poder anotar las tareas que tiene que realizar y conocer el estado y la prioridad de las mismas. Para poder solucionar el segundo problema, es decir, acotar responsabilidades y compartimentar el trabajo, las herramientas ETL tendrían que ser aplicaciones centralizadas que agrupasen los trabajos en áreas de trabajo a modo de directorios a las que sólo pudiesen acceder usuarios registrados y, más concretamente, sólo a las áreas de trabajo en las que estuviesen autorizados a trabajar.

Por otra parte, resulta llamativo que las herramientas ETL desarrolladas como aplicaciones Web sean prácticamente residuales. Hace unos años esto sería entendible puesto que el intercambio de datos en redes de ordenadores como Internet resultaba deficiente y la experiencia del usuario con este tipo de aplicaciones no era positiva. Sin embargo, hoy en día el intercambio de

datos en las redes de ordenadores de cualquier compañía es muy rápido y seguro. Asimismo, la tendencia de los últimos años es trabajar desde la nube. Por ejemplo, hace unos años no era habitual editar un documento de texto desde la nube, hoy en día es una práctica frecuente y cada vez más extendida. Por este motivo, a un usuario que trabaje con herramientas ETL no le va a suponer ningún rechazo trabajar desde un navegador web, es más, se va a sentir más cómodo. Por tanto, presentar una herramienta ETL como aplicación Web es viable y no debe entenderse como una tendencia o una moda, ya que hay que tener presente que este tipo de aplicaciones llevan intrínseco las siguientes ventajas: independencia del sistema operativo, acceso inmediato y desde cualquier lugar, soporte de múltiples usuarios concurrentes, actualizaciones sin requerir la intervención del usuario y mayor seguridad al evitar los problemas que puedan tener los ordenadores de los usuarios, por ejemplo, posibles virus.

Si se recuperan los objetivos marcados del presente trabajo, se puede determinar que han sido superados con éxito. Se ha estudiado qué es Big Data y el papel que juega en el rendimiento de una organización. Se ha hecho especial hincapié en explicar el proceso que se encarga del movimiento y transformación de datos, puesto que el objetivo principal de este proyecto ha sido estudiar las herramientas ETL que están destinadas a realizar esta labor y, al mismo tiempo, ofrecer una alternativa solvente que cubra las necesidades encontradas. El resultado ha sido una aplicación web que permite elaborar procesos ETL mediante programación por componentes, pudiendo gestionarlos y ejecutarlos. Más concretamente, en el trabajo con ficheros de tipo CSV permitiendo realizar tareas de mapeo de campos y filtrado de registros sobre el flujo de datos. Se ha comprobado que toda la funcionalidad demandada ha sido incluida mediante una evaluación con una batería de pruebas exhaustiva.

Por consiguiente, se pone de manifiesto que la aplicación desarrollada es competente, funcional y se ajusta a los objetivos marcados al comienzo del proyecto.

6.2. Trabajos futuros

Las líneas de trabajo futuras deben ir encaminadas a integrar aquellas funcionalidades que incorporan las herramientas ETL actuales en la aplicación gestada en este proyecto, no incluidas al conllevar años de desarrollo. De esta manera, se conseguirá una herramienta ETL muy potente por la que las empresas estarán dispuestas a invertir su capital. Ya que, por un lado, incluirá las funcionalidades de las herramientas ETL actuales que las hacen tan potentes a la hora de elaborar procesos ETL y, por otro, incorporará las carencias que éstas presentan en cuanto a la gestión del trabajo y la usabilidad resueltas en la aplicación desarrollada en este trabajo.

A continuación, se detallan los puntos que deben constituir la base de las futuras mejoras:

- Aumentar el número de componentes soportados

Esto se traduce en que se deben incluir progresivamente componentes que soporten más tipos de fuentes de entrada y de salida de datos, léase bases de datos como MySQL u Oracle, ficheros de tipo XML y demás fuentes de datos existentes. Asimismo, debe ampliarse el número de operaciones de transformación soportadas, como eliminar columnas de datos o bifurcar el flujo de datos atendiendo a ciertas condiciones.

- Permitir descargar los procesos ETL

Esta funcionalidad ha sido planteada en la presente memoria, pero los requisitos que se obtuvieron al respecto fueron rechazados al implicar un tiempo de desarrollo superior al establecido para este trabajo.

Esta funcionalidad consiste en descargar un proceso generado desde la aplicación en el equipo del usuario con todas las librerías requeridas para que pueda ser ejecutado. Esto permitiría producir un trabajo autónomo que no dependiese de la aplicación desde la que se realizó.

- Sistema de logs más detallados

Actualmente, se muestra una descripción del posible error producido, la fecha y el estado de la ejecución de un trabajo ETL. En un futuro, sería conveniente que se arrojase información sobre cada componente. Es decir, si se produjera un error en una determinada zona del flujo de trabajo, el usuario supiese qué componente ha sido el causante de dicho error y contase con un sistema de debug que le facilitase la tarea de poder localizarlo y subsanarlo.

- Mejorar los tiempos de ejecución

Aún consiguiendo resultados competitivos, la ejecución de un mismo trabajo con la herramienta ETL desarrollada en este proyecto es más lenta que la realizada con una de las herramientas ETL más punteras del mercado. Para mejorar los tiempos de ejecución se deben elaborar algoritmos más eficientes aplicados a las operaciones de transformación.

6.3. Problemas encontrados

El estudio que se inició a raíz de la necesidad de traer grandes volúmenes de datos desde diversas fuentes en diversos formatos y diferentes niveles de calidad juntos y almacenarlos en un sistema de destino determinó que era necesario contar con herramientas que permitiesen realizar estas tareas. El primer problema surgió al intentar encontrar carencias en las actuales herramientas de este tipo lo suficientemente importantes como para que mereciese la pena desarrollar una aplicación que las cubriese. El número de herramientas ETL existentes, la variedad y los años de vida de algunas de ellas no invitaba a ser optimista al respecto. Sin embargo, comprender la finalidad de este tipo de herramientas y el estudio de las principales que ofrece el mercado permitieron alcanzar este objetivo.

Otro de los problemas encontrados fue tener que hacer frente al gran volumen de requisitos obtenido, hecho que supuso que tanto el diseño como la implementación de la herramienta conllevaran un gran esfuerzo y consumieran gran parte del tiempo dedicado al proyecto. Este número de requisitos tan elevado se debió a que se quiso que la aplicación gestada cubriese el máximo posible de carencias encontradas en las actuales herramientas ETL.

En cuanto al diseño, el principal problema vino a la hora de determinar cómo presentar el proceso ETL en la vista y cómo establecer la interacción del usuario con cada uno de sus componentes. En un principio se pensó en diseñar desde cero estos componentes y las líneas encargadas de representar visualmente sus conexiones, subestimando el esfuerzo que esto implicaría. El mayor problema no fue representar los componentes en pantalla, sino las conexiones entre ellos, teniendo que pensar en algoritmos que permitiesen trazar una línea desde una coordenada a otra. Además, estas líneas no podían ser estáticas, puesto que la idea

era que el usuario pudiera distribuir los componentes en el área de trabajo según su propio criterio. Llegados a este punto, se investigó si existía alguna librería que ayudase a la creación de gráficos y solucionase este problema. La investigación dio sus frutos en forma de una librería de gráficos de JavaScript llamada jsPlumb, que solucionaba perfectamente el problema. Los errores tienen que servir para hacer mejor las cosas en proyectos venideros y, en este caso, se ha aprendido que es más rentable gastar recursos en investigar tecnologías que resuelvan o traten problemas que haya que solventar que afrontar un trabajo sin tener ninguna referencia previa. En muchos casos no se tratará de emplear directamente estas tecnologías, siendo suficiente estudiarlas y aprender cómo resuelven esos problemas a encarar.

En cuanto a la implementación, además de tener que hacer frente al gran volumen de requisitos generado, el principal escollo fue implementar la estructura que mantiene en memoria un proceso ETL a medida que el usuario va realizándolo. Se pudo constatar que no era viable guardar cada cambio realizado por el usuario en base de datos, ya que ralentizaba el trabajo y requería deshacer los cambios en el caso de que el usuario saliese del área de elaboración del proceso ETL sin aplicar estos cambios. Este problema no se identificó durante el diseño, lo que provocó tener que hacerle frente durante la etapa de implementación.

Por último, como problema destacado, la duración del proyecto. Se considera que el proyecto gestado en este trabajo es ambicioso y que puede dar lugar a una herramienta que se ajuste a las necesidades reales de las personas dedicadas a la elaboración de procesos ETL. Para ello, faltaría poder igualar el paquete de componentes que ofrecen las herramientas ETL actuales, pero no ha sido posible al no disponer del tiempo que requeriría este desarrollo.

6.4. Opiniones personales

Antes de embarcarme en este proyecto trabajaba en el departamento de desarrollo de una consultora dedicada a la externalización de procesos de negocio de un banco y aún sigo haciéndolo en estos momentos en los que estoy escribiendo estas reflexiones.

Entre las tareas que tengo que realizar en mi trabajo se encuentra la de tener que elaborar procesos ETL. Estos procesos pueden ser tan simples como extraer datos de ficheros de tipo CSV de gran tamaño, transformarlos y cargarlos en tablas de una base de datos hasta procesos más complejos como la migración de una base de datos en Access a MariaDB con envío de correo con el detalle de la ejecución del proceso.

Para la realización de estos procesos empleo una de las herramientas ETL con licencia libre presentadas en esta memoria. Para ser más precisos, es empleada por todos los miembros del equipo que desempeñamos esta labor dentro del departamento. Esta herramienta nos ha permitido realizar todos los procesos ETL requeridos hasta la fecha, pero me he ido dando cuenta con el tiempo que la parte de la elaboración de procesos ETL es muy completa, pero en cuanto a la gestión presenta carencias como no tener en cuenta el grupo de trabajo que hace uso de este tipo de herramientas.

Esa sensación de que la herramienta ETL que empleo podía ser mejorada fue la chispa que terminó dando vida a este proyecto. Sin embargo, en aquel momento esta idea era sólo un esbozo y no sabía si sería capaz de darle forma y si las carencias encontradas tendrían el suficiente calado como para que mereciese la pena iniciar un trabajo.

Para determinar si la idea podía dar sus frutos en forma de proyecto fue crucial la intervención de mi tutor. Me recalcó la importancia de estudiar el marco en el que se desenvuelve esta tecnología y la importancia de realizar un estudio del arte incluyendo un análisis meticuloso de las herramientas ETL que ofrece el mercado. Gracias a ese trabajo preliminar pude constatar que aquellas carencias encontradas en la herramienta ETL con la que trabajo eran extrapolables al resto de herramientas estudiadas. Es decir, permiten elaborar procesos ETL muy complejos soportando una gran cantidad de fuentes de origen y de destino, pero no tienen en cuenta que cada vez es más común encontrar equipos especializados en producir procesos ETL dentro de los departamentos de desarrollo de las empresas y, como en cualquier otro equipo de trabajo, existe el problema de cómo optimizar las tareas y el de cómo acotar responsabilidades.

En cuanto al estudio del marco en el que se desenvuelve esta tecnología pude determinar que se trataba de Big Data. Este vocablo me resultaba familiar, pero nunca lo había estudiado en profundidad. Comprendí que por Big Data, de manera muy resumida, se entiende la captura y el manejo de gran cantidad de información. Durante la fase de documentación pude encontrar principalmente dos opiniones distintas: aquellos que consideran que Big Data será tarde o temprano prácticamente indispensable en las empresas con cierto peso y los que piensan que ya lo es. Mi opinión coincide con la segunda postura, es decir, a día de hoy esta tecnología ya forma parte de estas empresas.

A mi entender, aquellos que piensan que Big Data aún no es una realidad no tienen en cuenta que esta tecnología no sólo ha sido creada para dar solución al procesamiento de enormes conjuntos de datos no estructurados, sino que también trabaja con datos estructurados y semi-estructurados. Las grandes compañías sí que emplean esta tecnología para trabajar con datos no estructurados para, por ejemplo, llegar a analizar los gustos de sus potenciales clientes. Sin embargo, es cierto que en empresas de menor tamaño aún no se han explotado este tipo de datos al no considerar rentable la información que se pudiera obtener a partir de este tipo de datos. Pero, de ahí mi discrepancia, estas empresas más modestas sí que emplean esta tecnología aunque sólo sea para trabajar con datos estructurados, siendo muy extendido entre ellas utilizarla para almacenar sus datos históricos por varios años o para construir data warehouse y data mart cuyos datos serán transformados en información que ayude en la toma de decisiones.

Respecto a las herramientas ETL, son una realidad igual que Big Data. Forman parte de esta tecnología y se ocupan de su integración, concretamente, de la extracción de datos tanto homogénea como heterogénea, transformar los datos para su almacenamiento en el formato adecuado o estructura para las consultas y análisis y, además, cargar estos datos en el sistema de destino que corresponda. En mi opinión, tienen una esperanza de vida larga a pesar de los años que llevan en activo. Han sabido adaptarse a los nuevos tiempos y son capaces de trabajar con datos no estructurados soportando frameworks como Apache Hadoop que permite trabajar con miles de nodos y petabytes de datos y, por ejemplo, trabajar con el motor MapReduce. El único inconveniente que presentan es que han descuidado la parte de la gestión y el equipo de trabajo que hace uso de ellas, pero con las soluciones aportadas en este proyecto a ese respecto, esas perspectivas tan prometedoras seguirán intactas.

7. Bibliografía

[1]	Kimball, Ralph & Caserta, Joe. "The Data Warehouse ETL Toolkit". New Delhi Wiley India 2004. ISBN 8126505540; 9788126505548
[2]	Ángel Leo-Revilla, "Los procesos ETL, la base de los sistemas Big Data", 14 Junio 2013, [Online]. Disponible: https://momentotic.com/2013/06/14/los-procesos-etl-la-base-de-los-sistemas-bigdata
[3]	María Silvina Souza, "El estado del arte", [Online]. Disponible: http://perio.unlp.edu.ar/seminario/nivel2/nivel3/el%20estado%20del%20arte_silvina_souza.pdf
[4]	Judith Hurwitz, Alan Nugent, Fern Halper, Marcia Kaufman, "The Role of Traditional ETL in Big Data", [Online]. Disponible: http://www.dummies.com/programming/big-data/data-science/the-role-of-traditional-etl-in-big-data
[5]	Joe Caserta and Elliott Cordo, "Big ETL: The Next 'Big' Thing", 9 Febrero 2015, [Online]. Disponible: http://data-informed.com/big-etl-next-big-thing
[6]	Alex Rayón, "Herramientas ETL y su relevancia en la cadena de valor del dato", 16 Noviembre 2015, [Online]. Disponible: https://blogs.deusto.es/bigdata/herramientas-etl-y-su-relevancia-en-la-cadena-de-valor-del-dato
[7]	Roberto Espinosa, "Herramientas ETL. ¿Qué son, para qué valen? Productos más conocidos. ETL's Open Source.", 29 Diciembre 2009, [Online]. Disponible: https://churriwifi.wordpress.com/2009/12/29/11-herramientas-etl-%C2%BFque-son-para-que-valen-productos-mas-conocidos-etl%C2%B4s-open-source
[8]	PowerData, "Procesos ETL. La Base de la Inteligencia de Negocio", [Online]. Disponible: http://cdn2.hubspot.net/hub/239039/file-44151143-pdf/docs/PowerData_-
[9]	ETL-Tools, "Looking up data in Informatica", [Online]. Disponible: http://etl-tools.info/informatica/lookups.html
[10]	Sudhaa Gopinath, "Why Data Warehousing And Business Intelligence Go Hand-in-Hand", 14 Septiembre 2015, [Online]. Disponible: http://etl-tools.info/informatica/lookups.html
[11]	Rafael Espejo, "Gestión visual de proyectos: tableros Kanban", 20 Julio 2015, [Online]. Disponible: http://prevenblog.com/gestion-visual-proyectos-tableros-kanban
[12]	ETL Tools, "Extraction", [Online]. Disponible: http://www.etltools.org/extraction.html
[13]	ETL Tools, "Data Transformation", [Online]. Disponible: http://www.etltools.org/data-transformation.html
[14]	SAS, "Big Data History and Current Considerations", [Online]. Disponible: http://www.sas.com/en_th/insights/big-data/what-is-big-data.html
[15]	Kevin Normandeau, "Beyond Volume, Variety and Velocity is the Issue of Big Data Veracity", 12 Septiembre 2013, [Online]. Disponible: http://insidebigdata.com/2013/09/12/beyond-volume-variety-velocity-issue-big-data-veracity
[16]	Mediapost Group, "Las 5 Vs del Big Data", 25 Mayo 2016, [Online]. Disponible: http://www.mediapostgroup.es/blog/las-5-vs-del-big-data
[17]	Bernard Marr, "Big Data: The 5 Vs Everyone Must Know", 6 Marzo 2014, [Online]. Disponible: https://www.linkedin.com/pulse/20140306073407-64875646-big-data-the-5-vs-everyone-must-know
[18]	Wikipedia, "Big data", [Online]. Disponible: https://en.wikipedia.org/wiki/Big_data

[19]	Mark van Rijmenam, "Why The 3V's Are Not Sufficient To Describe Big Data", 7 Agosto 2016, [Online]. Disponible: https://datafloq.com/read/3vs-sufficient-describe-big-data/166
[20]	Diya Soubra, "The 3 Vs that define Big Data", 5 Julio 2012, [Online]. Disponible: http://www.datasciencecentral.com/forum/topics/the-3vs-that-define-big-data
[21]	Alicia Baró, "Las 5 Vs del Big Data en el Marketing Digital", 1 Julio 2016, [Online]. Disponible: http://blogueandoalos50.com/las-5-vs-del-big-data-en-el-marketing-digital
[22]	Doug Laney, "3D Data Management: Controlling Data Volume, Velocity and Variety", [Online]. Disponible: http://blogs.gartner.com/doug-laney/files/2012/01/ad949-3D-Data-Management-Controlling-Data-Volume-Velocity-and-Variety.pdf
[23]	El Mundo, "El volumen de datos almacenados en Internet será 44 veces mayor en 2020", [Online]. Disponible: http://www.elmundo.es/elmundo/2013/07/30/navegante/1375199227.html
[24]	IDC, "Worldwide Big Data and Business Analytics Revenues Forecast to Reach \$187 Billion in 2019, According to IDC", [Online]. Disponible: http://www.idc.com/getdoc.jsp?containerId=prUS41306516
[25]	Talend, "ETL for Analytics", [Online]. Disponible: http://www.talend.com/solutions/etl-analytics
[26]	Ddavalos, "En Informática ¿Qué es ETL?", 5 Octubre 2010, [Online]. Disponible: https://ddavalos99.wordpress.com/2010/10/05/en-informatica-%C2%BFque-es-etl/
[27]	GTI, "5 diferencias entre Big Data y Business Intelligence", 26 Febrero 2015, [Online]. Disponible: http://noticias.gti.es/productos/5-diferencias-entre-big-data-y-business-intelligence
[28]	Ángel Leo-Revilla, "La diferencia entre Big Data y Business Intelligence", 23 Diciembre 2013, [Online]. Disponible: https://momentotic.com/2013/12/23/la-diferencia-entre-bigdata-y-business-intelligence
[29]	Dataops, "What's the difference between Business Intelligence and Big Data?", 5 Octubre 2015, [Online]. Disponible: http://dataops.co/whats-the-difference-between-business-intelligence-and-big-data
[30]	Yinnavation, "¿Qué es SCM, CRM y ERP? ¿Cuál es su relación la innovación?", 16 Mayo 2009, [Online]. Disponible: http://yinnavation.blogspot.com.es/2009/05/que-es-scm-crm-y-erp-cual-es-su_15.html
[31]	Sinnexus, "Datamart", [Online]. Disponible: http://www.sinnexus.com/business_intelligence/datamart.aspx
[32]	Sinnexus, "Datawarehouse", [Online]. Disponible: http://www.sinnexus.com/business_intelligence/datawarehouse.aspx
[33]	Wikipedia, "Almacén de datos", [Online]. Disponible: https://es.wikipedia.org/wiki/Almac%C3%A9n_de_datos
[34]	Gustavo Hdz., "Data mart", 13 Junio 2012, [Online]. Disponible: http://es.slideshare.net/GustavoHernandez10/data-mart
[35]	Rapid Business Intelligence Success, "What is a Data Mart?", [Online]. Disponible: http://www.rapid-business-intelligence-success.com/data-mart.html
[36]	Buyto, "Almacenamiento de datos", [Online]. Disponible: http://www.buyto.es/general-business-intelligence/almacenamiento-de-datos-datawarehouse-datamart-en-business-intelligence
[37]	InetSoft, "InetSoft Webinar: The Definition of ETL and Its Advantages and Disadvantages", [Online]. Disponible: https://www.inetsoft.com/business/solutions/etl_definition_advantages_and_disadvantages/

[38]	Vincent McBurney, "Old ETL tools versus new ETL tools", 12 Septiembre 2008, [Online]. Disponible: http://it.toolbox.com/blogs/infosphere/old-etl-tools-versus-new-etl-tools-27151
[39]	Wikipedia, "Contenido libre", [Online]. Disponible: https://es.wikipedia.org/wiki/Contenido_libre
[40]	Wikipedia, "Software propietario", [Online]. Disponible: https://es.wikipedia.org/wiki/Software_propietario
[41]	PowerData, "Desafíos para los procesos y herramientas ETL", 11 Julio 2013, [Online]. Disponible: http://blog.powerdata.es/el-valor-de-la-gestion-de-datos/bid/289577/Desaf-os-para-los-procesos-y-herramientas-ETL
[42]	ETL tools, "ETL Tools Comparison", [Online]. Disponible: http://www.etltools.net/etl-tools-comparison.html
[43]	Roberto Marchetto, "Talend Open Studio vs Pentaho Kettle, a comparison", 14 Mayo 2011, [Online]. Disponible: http://www.robertomarchetto.com/talend_studio_vs_kettle_pentao_pdi_comparis_on
[44]	Víctor Javier Madrid, "Talend Open Studio (TOS) 4.0", 6 Mayo 2010, [Online]. Disponible: https://www.adictosaltrabajo.com/tutoriales/tos-4-0
[45]	Roberto Espinosa, "Comparativa ETL Talend vs Pentaho Data Integration (Kettle)", 1 Junio 2010, [Online]. Disponible: https://churriwifi.wordpress.com/2010/06/01/comparativa-talend-vs-kettle-pdi
[46]	Respinos Amilla, "Herramientas ETL. ¿Qué son, para qué valen? Productos más conocidos. ETL's Open Source.", 25 Febrero 2010, [Online]. Disponible: http://www.dataprix.com/blogs/respinosamilla/herramientas-etl-que-son-para-que-valen-productos-mas-conocidos-etl-s-open-sour
[47]	Wikipedia, "Extract, transform and load", [Online]. Disponible: https://es.wikipedia.org/wiki/Extract,_transform_and_load
	Talend Expert, "Pentaho VS Talend", 29 Mayo 2015, [Online]. Disponible: https://talendexpert.com/pentaho-vs-talend
[48]	Vincent McBurney, "Wiki Wednesday: comparing Talend and Pentaho Kettle open source ETL tools", 16 Mayo 2007, [Online]. Disponible: http://it.toolbox.com/blogs/infosphere/wiki-wednesday-comparing-talend-and-pentaho-kettle-open-source-etl-tools-16294
[49]	Apache, "Apache Oozie Workflow Scheduler for Hadoop", 17 Agosto 2016, [Online]. Disponible: https://oozie.apache.org/
[50]	Hortonworks, "Apache Oozie", [Online]. Disponible: http://hortonworks.com/apache/oozie
[51]	Json, "Introducing JSON", [Online]. Disponible: http://www.json.org/
[52]	Kasia Mikoluk, "JSON vs XML: Cómo JSON es Superior a XML", 31 Diciembre 2013, [Online]. Disponible: https://blog.udemy.com/json-vs-xml-como-json-es-superior-a-xml
[53]	Wikipedia, "JSON", [Online]. Disponible: https://es.wikipedia.org/wiki/JSON
[54]	Tutorialspoint, "WSDL-Introduction", [Online]. Disponible: http://www.tutorialspoint.com/wsdl/wsdl_introduction.htm
[55]	Wikipedia, "Servicio web", [Online]. Disponible: https://es.wikipedia.org/wiki/Servicio_web
[56]	JJAS, "Web Service: Definición, utilización y estructura del WSDL", 18 Enero 2012, [Online]. Disponible: http://programacion.jias.es/2012/01/web-service-definicion-utilizacion-estructura-del-wsdl

[57]	Julián Pérez Porto y Ana Gardey, "Definición de XML", 2013, [Online]. Disponible: http://definicion.de/xml
[58]	Wikipedia, "Extensible Markup Language", [Online]. Disponible: https://es.wikipedia.org/wiki/Extensible_Markup_Language
[59]	Jordi Torres, "Big Data (4): ¿Por qué las bases de datos relacionales no sirven para todo?", 14 Mayo 2013, [Online]. Disponible: http://www.tecnonews.info/opiniones/big_data_4_por_que_las_bases_de_datos_relacionales_no_sirven_para_todo
[60]	Carlos Paramio, "El concepto NoSQL, o cómo almacenar tus datos en una base de datos no relacional", 26 Abril 2011, [Online]. Disponible: http://www.genbetadev.com/bases-de-datos/el-concepto-nosql-o-como-almacenar-tus-datos-en-una-base-de-datos-no-relacional
[61]	Pandorafms, "NOSQL vs SQL. Diferencias y cuando elegir cada una", 18 Noviembre 2015, [Online]. Disponible: http://blog.pandorafms.org/es/nosql-vs-sql-diferencias-y-cuando-elegir-cada-una/
[62]	Irma Rivas, "Características de aplicaciones web", [Online]. Disponible: http://estudiantealdeunare3irmadj.blogspot.com.es/p/caracteristicas-de-aplicaciones-web.html
[63]	Digital Marketplace, "Informatica Cloud Data Integration / ETL (DI/ETL)", [Online]. Disponible: https://www.digitalmarketplace.service.gov.uk/g-cloud/services/7358380159843866
[64]	G2CROWD, "Informatica Cloud Reviews", [Online]. Disponible: https://www.g2crowd.com/products/informatica-cloud/reviews
[65]	Informatica, "The cloud challenge", [Online]. Disponible: https://www.informatica.com/products/cloud-integration.html
[66]	Wikipedia, "Informatica", [Online]. Disponible: https://en.wikipedia.org/wiki/Informatica
[67]	Richard Seroter, "ETL in the Cloud with Informatica: Part 1 – Sending File Data to Dynamics CRM Online", 26 Marzo 2012, [Online]. Disponible: https://seroter.wordpress.com/2012/03/26/etl-in-the-cloud-with-informatica-part-1-sending-file-data-to-dynamics-crm-online
[68]	PowerData, "Las ventajas del Cloud Integration Service de Informatica", 16 Septiembre 2015, [Online]. Disponible: http://blog.powerdata.es/el-valor-de-la-gestion-de-datos/bid/407060/Las-ventajas-del-Cloud-Integration-Service-de-Informatica
[69]	Jane Randolph, "Installing an Informatica Cloud Secure Agent for Data Wizard", 27 Abril 2016, [Online]. Disponible: https://network.informatica.com/docs/DOC-16588
[70]	Spring, "Introduction to Spring Web MVC framework", [Online]. Disponible: http://docs.spring.io/spring/docs/current/spring-framework-reference/html/mvc.html
[71]	Tutorialspoint, "Spring – MVC Framework Tutorial", [Online]. Disponible: http://www.tutorialspoint.com/spring/spring_web_mvc_framework.htm
[72]	Wikipedia, "Single-page application", [Online]. Disponible: https://es.wikipedia.org/wiki/Single-page_application
[73]	OBS Business School, "¿Qué es un diagrama de Gantt y para qué sirve?", [Online]. Disponible: http://www.obs-edu.com/es/blog-project-management/diagramas-de-gantt/que-es-un-diagrama-de-gantt-y-para-que-sirve
[74]	Wikipedia, "Diagrama de Gantt", [Online]. Disponible: https://es.wikipedia.org/wiki/Diagrama_de_Gantt

[75]	Wikipedia, "Prueba unitaria", [Online]. Disponible: https://es.wikipedia.org/wiki/Prueba_unitaria
[76]	Wikipedia, "Pruebas funcionales", [Online]. Disponible: https://es.wikipedia.org/wiki/Pruebas_funcionales
[77]	UNAD, "El modelo lineal secuencial", [Online]. Disponible: http://datateca.unad.edu.co/contenidos/301404/301404_ContentadoEnLinea/leccin_11_el_modelo_lineal_secuencial.html
[78]	Ingenexescom, "Modelo Lineal secuencial", 7 Febrero 2012, [Online]. Disponible: http://ingenexescom.blogspot.com.es/2012/02/modelo-lineal-secuencial.html
[79]	Wikipedia, "Caso de uso", [Online]. Disponible: https://es.wikipedia.org/wiki/Caso_de_uso
[80]	Jordisan, "¿Qué es un 'framework'?", 29 Septiembre 2006, [Online]. Disponible: http://jordisan.net/blog/2006/que-es-un-framework
[81]	Websystique, "Spring 4 MVC+Hibernate 4+MySQL+Maven integration example using annotations", 30 Julio 2016, [Online]. Disponible: http://websystique.com/springmvc/spring-4-mvc-and-hibernate4-integration-example-using-annotations/
[82]	Websystique, "Spring MVC 4 + Spring Security 4 + Hibernate Example", 30 Julio 2016, [Online]. Disponible: http://websystique.com/springmvc/spring-mvc-4-and-spring-security-4-integration-example/
[83]	Wikipedia, "Hibernate", [Online]. Disponible: https://es.wikipedia.org/wiki/Hibernate
[84]	CakePHP, "Entendiendo el Modelo – Vista - Controlador", [Online]. Disponible: http://book.cakephp.org/2.0/es/cakephp-overview/understanding-model-view-controller.html
[85]	Alejandro Pérez García, "Spring + Hibernate + Anotaciones = Desarrollo Rápido en Java", 9 Mayo 2008, [Online]. Disponible: https://www.adictosaltrabajo.com/tutoriales/desarrollo-rapido-java/
[86]	Mkyong, "Spring 4 MVC Ajax Hello World Example", 20 Octubre 2015, [Online]. Disponible: http://www.mkyong.com/spring-mvc/spring-4-mvc-ajax-hello-world-example/
[87]	Wikipedia, "Programación orientada a componentes", [Online]. Disponible: https://es.wikipedia.org/wiki/Programaci%C3%B3n_orientada_a_componentes
[88]	jsPlumb, "Community Edition Documentation", [Online]. Disponible: https://jsplumbtoolkit.com/community/doc/home.html

Tabla 37 - Tabla de Bibliografía

Anexo I. Requisitos

Identificador	RRG-01		Categoría	Acceso al sistema	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	El acceso a la herramienta ETL Web se hará mediante <i>nick</i> y <i>clave</i> .				
Requisitos relacionados					

Tabla 38 - Requisito RRG-01

Identificador	RRG-02		Categoría	Reinicio de clave de usuario	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	La solicitud de reinicio de contraseña se hará mediante <i>correo electrónico</i> .				
Requisitos relacionados					

Tabla 39 - Requisito RRG-02

Identificador	RRG-03		Categoría	Reinicio de clave de usuario	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Descripción	El cambio de clave se hará desde la URL de cambio de clave recibida por correo indicando la nueva clave del usuario de la aplicación.				
Requisitos relacionados		RRG-02			

Tabla 40 - Requisito RRG-03

Identificador	RFAA-04	Categoría	Gestión áreas de trabajo
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá añadir un nuevo área de trabajo.		
Requisitos relacionados			

Tabla 41 - Requisito RFAA-04

Identificador	RRAA-05	Categoría	Gestión áreas de trabajo
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema incluirá un panel para añadir un nuevo área de trabajo. Incluirá un campo para escribir el nombre del área de trabajo.		
Requisitos relacionados	RFAA-04		

Tabla 42 - Requisito RRAA-05

Identificador	RRAA-06	Categoría	Gestión áreas de trabajo
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá acceder al panel para añadir un nuevo área de trabajo a través del botón "Añadir área".		
Requisitos relacionados	RFAA-04, RRAA-05		

Tabla 43 - Requisito RRAA-06

Identificador	RRAA-07	Categoría	Gestión áreas de trabajo
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	La inclusión de un nuevo área de trabajo se hará escribiendo el nombre del área de trabajo en el campo de texto del panel para añadir un nuevo área de trabajo y pulsando el botón "Añadir". Mostrará un mensaje al usuario para que sepa que se ha efectuado la acción.		
Requisitos relacionados	RFAA-04, RRAA-05, RRAA-06		

Tabla 44 - Requisito RRAA-07

Identificador	RRAA-08	Categoría	Gestión áreas de trabajo
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema actualizará el listado de áreas de trabajo tras añadir una nueva.		
Requisitos relacionados	RFAA-04, RRAA-05, RRAA-06, RRAA-07		

Tabla 45 - Requisito RRAA-08

Identificador	RRAA-09	Categoría	Gestión áreas de trabajo
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El campo del nombre será obligatorio para poder añadir un nuevo área de trabajo.		
Requisitos relacionados	RFAA-04, RRAA-05, RRAA-06, RRAA-07		

Tabla 46 - Requisito RRAA-09

Identificador	RRAA-10	Categoría	Gestión áreas de trabajo
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	No podrá añadirse un área de trabajo con el mismo nombre que otra existente.		
Requisitos relacionados	RFAA-04, RRAA-05, RRAA-06, RRAA-07		

Tabla 47 - Requisito RRAA-10

Identificador	RFAA-11	Categoría	Gestión áreas de trabajo
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá visualizar las áreas de trabajo paginadas.		
Requisitos relacionados			

Tabla 48 - Requisito RFAA-11

Identificador	RRAA-12	Categoría	Gestión áreas de trabajo
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema mostrará las áreas de trabajo por filas. Cada fila de un área de trabajo mostrará el nombre del área de trabajo y el número de usuarios que tenga asociados.		
Requisitos relacionados	RFAA-11		

Tabla 49 - Requisito RRAA-12

Identificador	RFAA-13	Categoría	Gestión áreas de trabajo
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá borrar un área de trabajo.		
Requisitos relacionados			

Tabla 50 - Requisito RFAA-13

Identificador	RRAA-14	Categoría	Gestión áreas de trabajo
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	La eliminación de un área de trabajo se hará pulsando el botón con el icono de una papelera asociado a cada área de trabajo. Requerirá confirmación.		
Requisitos relacionados	RFAA-13		

Tabla 51 - Requisito RRAA-14

Identificador	RRAA-15	Categoría	Gestión áreas de trabajo
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema actualizará el listado de áreas de trabajo tras eliminar un área de trabajo.		
Requisitos relacionados	RFAA-13, RRAA-14		

Tabla 52 - Requisito RRAA-15

Identificador	RFAA-16	Categoría	Gestión áreas de trabajo
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá buscar un área de trabajo existente.		
Requisitos relacionados			

Tabla 53 - Requisito RFAA-16

Identificador	RRAA-17	Categoría	Gestión áreas de trabajo
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	La búsqueda se realizará seleccionando un área de trabajo existente en el campo de autocompletar con el símbolo de una lupa que incluirá el sistema.		
Requisitos relacionados	RFAA-16		

Tabla 54 - Requisito RRAA-17

Identificador	RRAA-18	Categoría	Gestión áreas de trabajo
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema mostrará una única fila con los datos del área de trabajo buscada.		
Requisitos relacionados	RFAA-16, RRAA-17		

Tabla 55 - Requisito RRAA-18

Identificador	RFAA-19	Categoría	Gestión áreas de trabajo
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá vincular un usuario existente a un área de trabajo.		
Requisitos relacionados			

Tabla 56 - Requisito RFAA-19

Identificador	RRAA-20	Categoría	Gestión áreas de trabajo
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema incluirá un panel para gestionar los usuarios vinculados a un área de trabajo.		
Requisitos relacionados	RFAA-19		

Tabla 57 - Requisito RRAA-20

Identificador	RRAA-21	Categoría	Gestión áreas de trabajo
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El panel para gestionar los usuarios vinculados a un área de trabajo incluirá un apartado para vincular un usuario al área de trabajo. Incluirá un campo autocompletar para buscar un usuario y un botón "Añadir" para vincular usuarios.		
Requisitos relacionados	RFAA-19, RRAA-20		

Tabla 58 - Requisito RRAA-21

Identificador	RRAA-22	Categoría	Gestión áreas de trabajo
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	La vinculación de un usuario existente a un área de trabajo se hará seleccionando el nombre del usuario del campo autocompletar y pulsando el botón "Añadir" desde el panel de gestión de usuarios.		
Requisitos relacionados	RFAA-19, RRAA-20, RRAA-21		

Tabla 59 - Requisito RRAA-22

Identificador	RRAA-23	Categoría	Gestión áreas de trabajo
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema actualizará el listado de usuarios de un área de trabajo tras añadir uno nuevo.		
Requisitos relacionados	RFAA-19, RRAA-20, RRAA-21, RRAA-22		

Tabla 60 - Requisito RRAA-23

Identificador	RFAA-24	Categoría	Gestión áreas de trabajo
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá visualizar los usuarios pertenecientes a un área de trabajo.		
Requisitos relacionados			

Tabla 61 - Requisito RFAA-24

Identificador	RRAA-25	Categoría	Gestión áreas de trabajo
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El panel para gestionar los usuarios vinculados a un área de trabajo incluirá un apartado para visualizar el listado de usuarios vinculados. Cada fila representará un usuario e incluirá el nombre del usuario vinculado.		
Requisitos relacionados	RFAA-24		

Tabla 62 - Requisito RRAA-25

Identificador	RFAA-26	Categoría	Gestión áreas de trabajo
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá desvincular un usuario de un área de trabajo.		
Requisitos relacionados			

Tabla 63 - Requisito RFAA-26

Identificador	RRAA-27	Categoría	Gestión áreas de trabajo
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	La desvinculación se hará a través del botón con el icono de una papelera asociado a cada usuario en el panel de gestión de usuarios de un área de trabajo. Requerirá confirmación.		
Requisitos relacionados	RFAA-26		

Tabla 64 - Requisito RRAA-27

Identificador	RRAA-28	Categoría	Gestión áreas de trabajo
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema actualizará el listado de usuarios de un área de trabajo tras desvincular un usuario.		
Requisitos relacionados	RFAA-26, RRAA-27		

Tabla 65 - Requisito RRAA-28

Identificador	RFAU-29	Categoría	Gestión de usuarios
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá añadir un nuevo usuario.		
Requisitos relacionados			

Tabla 66 - Requisito RFAU-29

Identificador	RRAU-30	Categoría	Gestión de usuarios
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema incluirá un panel para añadir un nuevo usuario. Incluirá un campo para el <i>nick</i> del usuario, un campo para el nombre del usuario, un campo para el correo y campos para elegir los roles del usuario.		
Requisitos relacionados	RFAU-29		

Tabla 67 - Requisito RRAU-30

Identificador	RRAU-31	Categoría	Gestión de usuarios
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá acceder al panel para añadir un nuevo área de trabajo a través del botón "Añadir área".		
Requisitos relacionados	RFAU-29, RRAU-30		

Tabla 68 - Requisito RRAU-31

Identificador	RRAU-32	Categoría	Gestión de usuarios
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	La inclusión de un nuevo usuario se hará escribiendo el <i>nick</i> , el nombre, el correo y el rol del usuario en el panel para añadir un nuevo usuario y pulsando el botón "Añadir". Mostrará un mensaje al usuario para que sepa que se ha efectuado la acción.		
Requisitos relacionados	RFAU-29, RRAU-30, RRAU-31		

Tabla 69 - Requisito RRAU-32

Identificador	RRAU-33	Categoría	Gestión de usuarios
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema actualizará el listado de usuarios tras añadir uno nuevo.		
Requisitos relacionados	RFAU-29, RRAU-30, RRAU-31, RRAU-32		

Tabla 70 - Requisito RRAU-33

Identificador	RRAU-34	Categoría	Gestión de usuarios
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	No podrá añadirse un usuario con el mismo <i>nick</i> y/o correo que uno existente.		
Requisitos relacionados	RFAU-29, RRAU-30, RRAU-31, RRAU-32		

Tabla 71 - Requisito RRAU-34

Identificador	RRAU-35	Categoría	Gestión de usuarios
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	Para añadir un usuario todos los campos serán obligatorios y al menos deberá tener un rol.		
Requisitos relacionados	RFAU-29, RRAU-30, RRAU-31, RRAU-32		

Tabla 72 - Requisito RRAU-35

Identificador	RRAU-36	Categoría	Gestión de usuarios
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	Un usuario podrá tener el rol de administrador y/o el de usuario.		
Requisitos relacionados	RFAU-29, RRAU-30, RRAU-31, RRAU-32		

Tabla 73 - Requisito RRAU-36

Identificador	RFAU-37	Categoría	Gestión de usuarios
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá modificar un usuario.		
Requisitos relacionados			

Tabla 74 - Requisito RFAU-37

Identificador	RRAU-38	Categoría	Gestión de usuarios
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema incluirá un panel para editar un usuario. Incluirá campos para modificar el nombre, los roles y el estado.		
Requisitos relacionados	RFAU-37		

Tabla 75 - Requisito RRAU-38

Identificador	RRAU-39	Categoría	Gestión de usuarios
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá acceder al panel para editar un usuario a través del botón con el icono de un lápiz asociado a cada usuario.		
Requisitos relacionados	RFAU-37, RRAU-38		

Tabla 76 - Requisito RRAU-39

Identificador	RRAU-40	Categoría	Gestión de usuarios
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	La modificación de un usuario se hará introduciendo el nuevo nombre del usuario, los nuevos roles y el nuevo estado y pulsando el botón "Editar". Mostrará un mensaje al usuario para que sepa que se ha efectuado la acción.		
Requisitos relacionados	RFAU-37, RRAU-38		

Tabla 77 - Requisito RRAU-40

Identificador	RRAU-41	Categoría	Gestión de usuarios
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	No podrá editarse un usuario que no incluya al menos un rol y cuyo nombre esté vacío.		
Requisitos relacionados	RFAU-37, RRAU-38, RRAU-40		

Tabla 78 - Requisito RRAU-41

Identificador	RFAU-42	Categoría	Gestión de usuarios
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá buscar un usuario.		
Requisitos relacionados			

Tabla 79 - Requisito RFAU-42

Identificador	RRAU-43	Categoría	Gestión de usuarios
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	La búsqueda se realizará seleccionando un usuario existente en el campo de autocompletar con el símbolo de una lupa que incluirá el sistema.		
Requisitos relacionados	RFAU-42		

Tabla 80 - Requisito RRAU-43

Identificador	RRAU-44	Categoría	Gestión de usuarios
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema mostrará una única fila con los datos del usuario buscado.		
Requisitos relacionados	RFAU-42, RRAU-43		

Tabla 81 - Requisito RRAU-44

Identificador	RFAU-45	Categoría	Gestión de usuarios
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá filtrar usuarios.		
Requisitos relacionados			

Tabla 82 - Requisito RFAU-45

Identificador	RRAU-46	Categoría	Gestión de usuarios
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El filtrado de usuarios se realizará a través de los campos de selección de usuarios activos (activos o no activos) y los campos de selección de usuarios con determinados roles (Administrador, Usuario o Todos).		
Requisitos relacionados	RFAU-45		

Tabla 83 - Requisito RRAU-46

Identificador	RFAU-47	Categoría	Gestión de usuarios
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá vincular un usuario a un área de trabajo existente.		
Requisitos relacionados			

Tabla 84 - Requisito RFAU-47

Identificador	RRAU-48	Categoría	Gestión de usuarios
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema incluirá un panel para gestionar las áreas de trabajo a las que pertenece un usuario.		
Requisitos relacionados	RFAU-47		

Tabla 85 - Requisito RRAU-48

Identificador	RRAU-49	Categoría	Gestión de usuarios
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El panel para gestionar las áreas de trabajo a las que pertenece un usuario incluirá un apartado para vincular el usuario a un área de trabajo. Incluirá un campo autocompletar para buscar un área de trabajo y un botón "Añadir" para vincular áreas de trabajo.		
Requisitos relacionados	RFAU-47, RRAU-48		

Tabla 86 - Requisito RRAU-49

Identificador	RRAU-50	Categoría	Gestión de usuarios
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	La vinculación del usuario a un área de trabajo existente se hará seleccionando el nombre del área de trabajo del campo autocompletar y pulsando el botón "Añadir" desde el panel de gestión de áreas de trabajo.		
Requisitos relacionados	RFAU-47, RRAU-48, RRAU-49		

Tabla 87 - Requisito RRAU-50

Identificador	RRAU-51	Categoría	Gestión de usuarios
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema actualizará el listado de áreas de trabajo a las que pertenezca el usuario tras añadir una nueva.		
Requisitos relacionados	RFAU-47, RRAU-48, RRAU-49, RRAU-50		

Tabla 88 - Requisito RRAU-51

Identificador	RFAU-52	Categoría	Gestión de usuarios
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá visualizar las áreas de trabajo a las que pertenece un usuario.		
Requisitos relacionados			

Tabla 89 - Requisito RFAU-52

Identificador	RRAU-53	Categoría	Gestión de usuarios
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El panel para gestionar las áreas de trabajo a las que pertenezca un usuario incluirá un apartado para visualizar el listado de áreas de trabajo vinculadas. Cada fila representará un área de trabajo e incluirá el nombre del área de trabajo vinculada.		
Requisitos relacionados	RFAU-52		

Tabla 90 - Requisito RRAU-53

Identificador	RFAU-54	Categoría	Gestión de usuarios
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá desvincular un usuario de un área de trabajo.		
Requisitos relacionados			

Tabla 91 - Requisito RFAU-54

Identificador	RRAU-55	Categoría	Gestión de usuarios
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	La desvinculación se hará a través del botón con el icono de una papelera asociado a cada área de trabajo en el panel de gestión de áreas de trabajo de un usuario. Requerirá confirmación.		
Requisitos relacionados	RFAU-54		

Tabla 92 - Requisito RRAU-55

Identificador	RRAU-56	Categoría	Gestión de usuarios
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema actualizará el listado de áreas de trabajo de un usuario tras desvincular un área de trabajo.		
Requisitos relacionados	RFAU-54, RRAU-55		

Tabla 93 - Requisito RRAU-56

Identificador	RFTP-57	Categoría	Planificación de trabajo
Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá añadir una etiqueta descriptiva de una actividad de trabajo a realizar.		
Requisitos relacionados			

Tabla 94 - Requisito RFTP-57

Identificador	RRTP-58	Categoría	Planificación de trabajo
Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema incluirá un panel para añadir una nueva etiqueta. Incluirá un campo para escribir el contenido de la etiqueta.		
Requisitos relacionados	RFTP-57		

Tabla 95 - Requisito RRTP-58

Identificador	RRTP-59	Categoría	Planificación de trabajo
Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	La inclusión de una nueva etiqueta se hará escribiendo el contenido en el campo de texto del panel para añadir una nueva etiqueta y pulsando el botón "Añadir".		
Requisitos relacionados	RFTP-57, RRTP-58		

Tabla 96 - Requisito RRTP-59

Identificador	RRTP-60	Categoría	Planificación de trabajo
Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema incluirá una etiqueta añadida en la columna de estado "Entrada".		
Requisitos relacionados	RFTP-57, RRTP-58, RRTP-59		

Tabla 97 - Requisito RRTP-60

Identificador	RFTP-61	Categoría	Planificación de trabajo
Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá visualizar el tablero con las etiquetas descriptivas de las actividades de trabajo del usuario de la aplicación.		
Requisitos relacionados			

Tabla 98 - Requisito RFTP-61

Identificador	RRTP-62	Categoría	Planificación de trabajo
Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema incluirá cuatro columnas, cada una representará un estado de trabajo: "Entrada", "Pendiente", "En curso" y "Hecho". Mostrará las etiquetas repartidas por las columnas según los estados en las que se encuentren.		
Requisitos relacionados	RFTP-61		

Tabla 99 - Requisito RRTP-62

Identificador	RFTP-63	Categoría	Planificación de trabajo
Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá modificar el estado de una etiqueta descriptiva de una actividad de trabajo a realizar.		
Requisitos relacionados			

Tabla 100 - Requisito RFTP-63

Identificador	RRTP-64	Categoría	Planificación de trabajo
Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El cambio de estado de una etiqueta se hará arrastrando y soltando la etiqueta de una columna de estado a otra. Existirán cuatro columnas de estado: "Entrada", "Pendiente", "En curso" y "Hecho".		
Requisitos relacionados	RFTP-63		

Tabla 101 - Requisito RRTP-64

Identificador	RFTP-65	Categoría	Planificación de trabajo
Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá borrar una etiqueta descriptiva de una actividad de trabajo a realizar existente.		
Requisitos relacionados			

Tabla 102 - Requisito RFTP-65

Identificador	RRTP-66	Categoría	Planificación de trabajo
Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	La eliminación de una etiqueta se hará pulsando el botón con el icono de una papelera asociado a cada etiqueta. No requerirá confirmación.		
Requisitos relacionados	RFTP-65		

Tabla 103 - Requisito RRTP-66

Identificador	RFTP-67	Categoría	Planificación de trabajo
Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá guardar los cambios efectuados en el tablero de planificación de trabajo.		
Requisitos relacionados			

Tabla 104 - Requisito RFTP-67

Identificador	RRTP-68	Categoría	Planificación de trabajo
Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El almacenamiento de los cambios efectuados en el tablero de planificación de trabajo se realizará pulsando el botón “Guardar” que estará ubicado en la parte superior del tablero. Mostrará un mensaje al usuario para que sepa que se ha efectuado la acción.		
Requisitos relacionados	RFTP-67		

Tabla 105 - Requisito RRTP-68

Identificador	RFTP-69	Categoría	Planificación de trabajo
Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá restaurar el tablero de planificación de trabajo con la última copia guardada.		
Requisitos relacionados			

Tabla 106 - Requisito RFTP-69

Identificador	RRTP-70	Categoría	Planificación de trabajo
Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	La restauración del tablero de planificación de trabajo se realizará pulsando el botón “Refrescar” que estará ubicado en la parte superior del tablero. Recuperará el tablero desde la última vez que se guardase.		
Requisitos relacionados	RFTP-69		

Tabla 107 - Requisito RRTP-70

Identificador	RFTE-71	Categoría	Programación de trabajos
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá visualizar el listado de programaciones de trabajos ETL.		
Requisitos relacionados			

Tabla 108 - Requisito RFTE-71

Identificador	RRTE-72	Categoría	Programación de trabajos
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema mostrará las programaciones por filas. Cada fila de programación de un trabajo mostrará el nombre del trabajo, el número de versión, la programación, la fecha de la última vez que se ejecutara, el estado de la última ejecución y la descripción de la última ejecución.		
Requisitos relacionados	RFTE-71		

Tabla 109 - Requisito RRTE-72

Identificador	RFTE-73	Categoría	Programación de trabajos
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá añadir una nueva programación de un trabajo ETL.		
Requisitos relacionados			

Tabla 110 - Requisito RFTE-73

Identificador	RRTE-74	Categoría	Programación de trabajos
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema incluirá un apartado para añadir una nueva programación que incluirá un campo para seleccionar el trabajo a programar, un selector de versión de trabajo del repositorio y un campo para definir el horario de la programación.		
Requisitos relacionados	RFTE-73		

Tabla 111 - Requisito RRTE-74

Identificador	RRTE-75	Categoría	Programación de trabajos
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	La inclusión de la programación de un trabajo se hará a través del botón con un símbolo de confirmación tras haber indicado el nombre del trabajo, la versión y el horario de programación en el apartado habilitado para añadir una nueva programación. Mostrará un mensaje al usuario para que sepa que se ha efectuado la acción.		
Requisitos relacionados	RFTE-73, RRTE-74		

Tabla 112 - Requisito RRTE-75

Identificador	RFTE-76	Categoría	Programación de trabajos
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá borrar una programación de un trabajo ETL.		
Requisitos relacionados			

Tabla 113 - Requisito RFTE-76

Identificador	RRTE-77	Categoría	Programación de trabajos
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	La eliminación de una programación se hará pulsando el botón con el icono de una papelera asociado a cada programación. No requerirá confirmación. Mostrará un mensaje al usuario para que sepa que se ha efectuado la acción.		
Requisitos relacionados	RFTE-76		

Tabla 114 - Requisitos RRTE-77

Identificador	RFTE-78	Categoría	Programación de trabajos
Prioridad	<input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja		
Riesgo	<input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá ejecutar un trabajo ETL.		
Requisitos relacionados			

Tabla 115 - Requisito RFTE-78

Identificador	RRTE-79		Categoría	Programación de trabajos	
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja		
Riesgo	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	La ejecución de la programación de un trabajo se hará pulsando el botón con el símbolo <i>play</i> asociado a cada programación. Mostrará un mensaje al usuario para que sepa que se ha efectuado la acción.				
Requisitos relacionados		RFTE-78			

Tabla 116 - Requisito RRTE-79

Identificador	RRTE-80		Categoría	Programación de trabajos	
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja		
Riesgo	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	El sistema mostrará la actualización de la fecha, el estado y la descripción de la última ejecución de la programación de un trabajo.				
Requisitos relacionados		RFTE-78, RRTE-79			

Tabla 117 - Requisito RRTE-80

Identificador	RFTE-81 [ELIMINADO]		Categoría	Programación de trabajos	
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja		
Riesgo	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja		
Verificabilidad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	El sistema permitirá descargar en local un trabajo ETL.				
Requisitos relacionados					

Tabla 118 - Requisito RFTE-81 [ELIMINADO]

Identificador	RRTE-82 [ELIMINADO]		Categoría	Programación de trabajos	
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja		
Riesgo	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja		
Verificabilidad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	La descarga en local de un trabajo se hará a través del botón con el símbolo de descarga asociado a cada programación.				
Requisitos relacionados		RFTE-81			

Tabla 119 - Requisito RRTE-82 [ELIMINADO]

Identificador	RRTE-83 [ELIMINADO]	Categoría	Programación de trabajos
Prioridad	<input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja		
Riesgo	<input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja		
Verificabilidad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	La descarga incluirá el programa y las librerías necesarios para poder ejecutar el proceso ETL.		
Requisitos relacionados	RFTE-81, RRTE-82		

Tabla 120 - Requisito RRTE-83 [ELIMINADO]

Identificador	RFTG-84	Categoría	Gestión de trabajos ETL
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá acceder a un área de trabajo al que pertenezca el usuario de la aplicación.		
Requisitos relacionados			

Tabla 121 - Requisito RFTG-84

Identificador	RRTG-85	Categoría	Gestión de trabajos ETL
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema incluirá una vista para poder acceder a un área de trabajo. Incluirá un campo autocompletar con el listado de áreas de trabajo a las que pertenezca el usuario de la aplicación y un botón para acceder al área de trabajo seleccionada.		
Requisitos relacionados	RFTG-84		

Tabla 122 - Requisito RRTG-85

Identificador	RRTG-86	Categoría	Gestión de trabajos ETL
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El acceso a un área de trabajo se realizará seleccionando un área de trabajo del campo autocompletar de áreas de trabajo y pulsando el botón "Seleccionar".		
Requisitos relacionados	RFTG-84, RRTG-85		

Tabla 123 - Requisito RRTG-86

Identificador	RRTG-87		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Descripción	El sistema sólo mostrará las áreas de trabajo a las que el usuario de la aplicación se encuentre vinculado.				
Requisitos relacionados		RFTG-84, RRTG-85, RRTG-86			

Tabla 124 - Requisito RRTG-87

Identificador	RFTG-88		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	El sistema ofrecerá una vista que permitirá visualizar un área de trabajo.				
Requisitos relacionados		RFTG-84			

Tabla 125 - Requisito RFTG-88

Identificador	RFTG-89		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	El sistema permitirá visualizar un trabajo perteneciente a un área de trabajo del usuario de la aplicación.				
Requisitos relacionados		RFTG-88			

Tabla 126 - Requisito RFTG-89

Identificador	RRTG-90		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Descripción	El sistema mostrará el nombre y la descripción del trabajo de un área de trabajo. Así como si el usuario de la aplicación tuviera una copia de usuario si existiera una versión de repositorio del trabajo. Además, mostrará las versiones de repositorio del trabajo.				
Requisitos relacionados		RFTG-89			

Tabla 127 - Requisito RRTG-90

Identificador	RFTG-91	Categoría	Gestión de trabajos ETL
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá añadir un nuevo trabajo dentro de un área de trabajo del usuario de la aplicación.		
Requisitos relacionados	RFTG-89		

Tabla 128 - Requisito RFTG-91

Identificador	RRTG-92	Categoría	Gestión de trabajos ETL
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema incluirá un panel para añadir un nuevo trabajo. Incluirá un campo para el nombre del trabajo y un campo para la descripción del trabajo.		
Requisitos relacionados	RFTG-91		

Tabla 129 - Requisito RRTG-92

Identificador	RRTG-93	Categoría	Gestión de trabajos ETL
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá acceder al panel para añadir un nuevo trabajo a través del botón "Añadir trabajo".		
Requisitos relacionados	RFTG-91, RRTG-92		

Tabla 130 - Requisito RRTG-93

Identificador	RRTG-94	Categoría	Gestión de trabajos ETL
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	La inclusión de un nuevo trabajo se hará escribiendo el nombre y la descripción del trabajo en el panel para añadir un nuevo trabajo y pulsando el botón "Añadir". Mostrará un mensaje al usuario para que sepa que se ha efectuado la acción.		
Requisitos relacionados	RFTG-91, RRTG-92, RRTG-93		

Tabla 131 - Requisito RRTG-94

Identificador	RRTG-95	Categoría	Gestión de trabajos ETL
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	No podrá añadirse un trabajo con el mismo nombre que otro existente dentro del mismo área de trabajo.		
Requisitos relacionados	RFTG-91, RRTG-92, RRTG-93, RRTG-94		

Tabla 132 - Requisito RRTG-95

Identificador	RFTG-96	Categoría	Gestión de trabajos ETL
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá buscar un trabajo perteneciente a un área de trabajo del usuario de la aplicación.		
Requisitos relacionados	RFTG-89		

Tabla 133 - Requisito RFTG-96

Identificador	RRTG-97	Categoría	Gestión de trabajos ETL
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	La búsqueda se realizará seleccionando un trabajo existente dentro del área de trabajo en el campo autocompletar etiquetado como "Seleccione un trabajo de la lista".		
Requisitos relacionados	RFTG-96		

Tabla 134 - Requisito RRTG-97

Identificador	RFTG-98	Categoría	Gestión de trabajos ETL
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá borrar un trabajo perteneciente a un área de trabajo del usuario de la aplicación.		
Requisitos relacionados	RFTG-89		

Tabla 135 - Requisito RFTG-98

Identificador	RRTG-99	Categoría	Gestión de trabajos ETL
---------------	---------	-----------	-------------------------

Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Descripción	La eliminación de un trabajo se hará pulsando el botón “Eliminar trabajo” asociado al trabajo abierto. Requerirá confirmación. Mostrará un mensaje al usuario para que sepa que se ha efectuado la acción.		
Requisitos relacionados	RFTG-98		

Tabla 136 - Requisito RRTG-99

Identificador	RRTG-100	Categoría	Gestión de trabajos ETL
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Descripción	La eliminación de un trabajo incluirá la eliminación de las vinculaciones con los usuarios y la eliminación de sus trabajos.		
Requisitos relacionados	RFTG-98, RRTG-99		

Tabla 137 - Requisito RRTG-100

Identificador	RFTG-101	Categoría	Gestión de trabajos ETL
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Descripción	El sistema permitirá buscar una versión de repositorio de un trabajo.		
Requisitos relacionados	RFTG-89, RRTG-90		

Tabla 138 - Requisito RFTG-101

Identificador	RRTG-102	Categoría	Gestión de trabajos ETL
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Descripción	La selección de la versión de un trabajo se hará a través del campo selector etiquetado con “Seleccione una versión” que incluirá todas las versiones de repositorio de un trabajo.		
Requisitos relacionados	RFTG-101		

Tabla 139 - Requisito RRTG-102

Identificador	RFTG-103		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	El sistema permitirá visualizar la información de una versión de repositorio de un trabajo.				
Requisitos relacionados		RFTG-101			

Tabla 140 - Requisito RFTG-103

Identificador	RRTG-104		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	El sistema mostrará el número de versión y la descripción de la versión seleccionada.				
Requisitos relacionados		RFTG-103			

Tabla 141 - Requisito RRTG-104

Identificador	RFTG-105		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	El sistema permitirá borrar una versión de repositorio de un trabajo.				
Requisitos relacionados		RFTG-103			

Tabla 142 - Requisito RFTG-105

Identificador	RRTG-106		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	La eliminación de una versión de repositorio de un trabajo se hará a través del botón “Eliminar versión repositorio” asociado a la versión seleccionada. Requerirá confirmación.				
Requisitos relacionados		RFTG-105			

Tabla 143 - Requisito RRTG-106

Identificador	RFTG-107		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Descripción	El sistema permitirá acceder a una versión del trabajo del repositorio.				
Requisitos relacionados		RFTG-103			

Tabla 144 - Requisito RFTG-107

Identificador	RRTG-108		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Descripción	El acceso a una versión de un trabajo se hará a través del botón “Abrir trabajo repositorio” asociado a la versión seleccionada.				
Requisitos relacionados		RFTG-107			

Tabla 145 - Requisito RRTG-108

Identificador	RFTG-109		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	El sistema permitirá acceder a la copia del trabajo que tenga el usuario de la aplicación.				
Requisitos relacionados		RFTG-89			

Tabla 146 - Requisito RFTG-109

Identificador	RRTG-110		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Descripción	El acceso a la copia del usuario del trabajo se hará a través del botón “Abrir trabajo usuario” asociado al trabajo.				
Requisitos relacionados		RFTG-109			

Tabla 147 - Requisito RRTG-110

Identificador	RFTG-111	Categoría	Gestión de trabajos ETL
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema permitirá elaborar un proceso ETL dentro de un trabajo perteneciente a un área de trabajo del usuario de la aplicación.		
Requisitos relacionados	RFTG-107, RFTG-109		

Tabla 148 - Requisito RFTG-111

Identificador	RRTG-112	Categoría	Gestión de trabajos ETL
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	El sistema ofrecerá una vista desde la que realizar procesos ETL dentro de un trabajo.		
Requisitos relacionados	RFTG-111		

Tabla 149 - Requisito RRTG-112

Identificador	RRTG-113	Categoría	Gestión de trabajos ETL
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	La vista para realizar procesos ETL tendrá un panel de componentes que incluirá un botón "FicheroEntrada" para crear componentes de entrada de ficheros de tipo CSV, un botón "FicheroSalida" para crear componentes de salida de ficheros de tipo CSV, un botón "Mapeo" para eliminar campos del flujo de entrada y para cruzar ficheros de entrada y un botón "Filtro" para filtrar registros procedentes del flujo de datos recibido.		
Requisitos relacionados	RRTG-112		

Tabla 150 - Requisito RRTG-113

Identificador	RRTG-114	Categoría	Gestión de trabajos ETL
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	La vista para realizar procesos ETL incluirá un área donde colocar los componentes seleccionados del panel de componentes y sus conectores.		
Requisitos relacionados	RRTG-112		

Tabla 151 - Requisito RRTG-114

Identificador	RRTG-115		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	Para crear un componente habrá que seleccionar el botón del componente correspondiente y hacer clic en el área habilitada para colocar los componentes y sus conectores.				
Requisitos relacionados		RRTG-114			

Tabla 152 - Requisito RRTG-115

Identificador	RRTG-116		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	El contenedor que representa visualmente un componente incluirá el nombre del componente.				
Requisitos relacionados		RRTG-114, RRTG-115			

Tabla 153 - Requisito RRTG-116

Identificador	RRTG-117		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	El sistema incluirá un panel por cada tipo de componente para poder gestionar este componente.				
Requisitos relacionados		RRTG-114, RRTG-115			

Tabla 154 - Requisito RRTG-117

Identificador	RRTG-118		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Descripción	El acceso al panel de gestión de un componente se hará haciendo doble clic sobre el contenedor que represente el componente.				
Requisitos relacionados		RRTG-114, RRTG-115, RRTG-117			

Tabla 155 - Requisito RRTG-118

Identificador	RRTG-119	Categoría	Gestión de trabajos ETL
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	Un componente "FicheroEntrada" sólo se podrá conectar en su salida con un componente "Mapeo" o con un componente "FicheroSalida".		
Requisitos relacionados	RRTG-114, RRTG-115, RRTG-117		

Tabla 156 - Requisito RRTG-119

Identificador	RRTG-120	Categoría	Gestión de trabajos ETL
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	Un componente "Mapeo" sólo se podrá conectar en su salida con un componente "Filtro" o con un componente "FicheroSalida".		
Requisitos relacionados	RRTG-114, RRTG-115, RRTG-117		

Tabla 157 - Requisito RRTG-120

Identificador	RRTG-121	Categoría	Gestión de trabajos ETL
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	Un componente "Filtro" sólo se podrá conectar en su salida con un componente "FicheroSalida".		
Requisitos relacionados	RRTG-114, RRTG-115, RRTG-117		

Tabla 158 - Requisito RRTG-121

Identificador	RRTG-122	Categoría	Gestión de trabajos ETL
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja		
Descripción	Un componente "FicheroEntrada" no podrá recibir ningún conector de entrada.		
Requisitos relacionados	RRTG-114, RRTG-115, RRTG-117		

Tabla 159 - Requisito RRTG-122

Identificador	RRTG-123		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Descripción	Un componente “FicheroSalida” no podrá conectarse con ningún componente en su salida.				
Requisitos relacionados		RRTG-114, RRTG-115, RRTG-117			

Tabla 160 - Requisito RRTG-123

Identificador	RRTG-124		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	Un cambio en un componente provocará el reinicio de los campos, relaciones y restricciones mapeados en los componentes que estén conectados a partir de su salida.				
Requisitos relacionados		RRTG-114, RRTG-115, RRTG-117			

Tabla 161 - Requisito RRTG-124

Identificador	RRTG-125		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	El panel de gestión de un componente de tipo “FicheroEntrada” incluirá un área para conectar el componente en su salida, un área para mapear los campos, un área para visualizar los campos mapeados y un área para indicar datos del fichero.				
Requisitos relacionados		RRTG-114, RRTG-115, RRTG-117			

Tabla 162 - Requisito RRTG-125

Identificador	RRTG-126		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Descripción	El panel de gestión de un componente de tipo “Mapeo” incluirá un área para conectar el componente en su salida, un área para mapear los campos recibidos y un área para relacionar los campos de dos ficheros de entrada si los hubiere.				
Requisitos relacionados		RRTG-114, RRTG-115, RRTG-117			

Tabla 163 - Requisito RRTG-126

Identificador	RRTG-127		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	El panel de gestión de un componente de tipo “Filtro” incluirá un área para conectar el componente en su salida, un área para visualizar los campos recibidos y un área para filtrar los campos recibidos.				
Requisitos relacionados		RRTG-114, RRTG-115, RRTG-117			

Tabla 164 - Requisito RRTG-127

Identificador	RRTG-128		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	El panel de gestión de un componente de tipo “FicheroSalida” incluirá un área para visualizar los campos recibidos y un área para indicar datos del fichero.				
Requisitos relacionados		RRTG-114, RRTG-115, RRTG-117			

Tabla 165 - Requisito RRTG-128

Identificador	RRTG-129		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	La eliminación de un componente se hará desde su panel de gestión correspondiente pulsando el botón “Eliminar componente”.				
Requisitos relacionados		RRTG-114, RRTG-115, RRTG-117, RRTG-125, RRTG-126, RRTG-127, RRTG-128			

Tabla 166 - Requisito RRTG-129

Identificador	RRTG-130		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	Un componente no podrá ser eliminado si tiene un conector de entrada.				
Requisitos relacionados		RRTG-129			

Tabla 167 - Requisito RRTG-130

Identificador	RRTG-131		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Descripción	Conectar un componente en su salida con otro componente implicará la eliminación de la posible conexión que tuviera con otro componente en su salida previamente.				
Requisitos relacionados		RRTG-114, RRTG-115, RRTG-117			

Tabla 168 - Requisito RRTG-131

Identificador	RRTG-132		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	La conexión de un componente con otro en su salida se hará seleccionando un componente del campo de selección de conectores disponibles y pulsando el botón “Aceptar” dentro del área de conector en el panel de gestión del componente.				
Requisitos relacionados		RRTG-114, RRTG-115, RRTG-117, RRTG-125, RRTG-126, RRTG-127, RRTG-128			

Tabla 169 - Requisito RRTG-132

Identificador	RRTG-133		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	La visualización de los campos mapeados en un componente se hará desde el área de visualización de campos dentro del panel de gestión del componente.				
Requisitos relacionados		RRTG-114, RRTG-115, RRTG-117, RRTG-125, RRTG-126, RRTG-127, RRTG-128			

Tabla 170 - Requisito RRTG-133

Identificador	RRTG-134		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	La inclusión de un campo en un componente de tipo “FicheroEntrada” se hará indicando el nombre y el tipo de campo y pulsando el botón “Añadir” en el área de mapeo de campos dentro del panel de gestión del componente.				
Requisitos relacionados		RRTG-114, RRTG-115, RRTG-117, RRTG-125			

Tabla 171 - Requisito RRTG-134

Identificador	RRTG-135		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Descripción	Los tipos de campos permitidos serán 'Boolean', 'Date', 'Double', 'Float', 'Integer' y 'String'.				
Requisitos relacionados		RRTG-114, RRTG-115, RRTG-117, RRTG-125			

Tabla 172 - Requisito RRTG-135

Identificador	RRTG-136		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	La eliminación de un campo de un componente se hará pulsando el botón con el icono de una papelera asociado al campo en el área de visualización de campos dentro del panel de gestión del componente.				
Requisitos relacionados		RRTG-114, RRTG-115, RRTG-117, RRTG-125, RRTG-126			

Tabla 173 - Requisito RRTG-136

Identificador	RRTG-137		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	Sólo los componentes de tipo “FicheroEntrada” y “Mapeo” podrán eliminar campos que pasen por ellos.				
Requisitos relacionados		RRTG-136			

Tabla 174 - Requisito RRTG-137

Identificador	RRTG-138		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	La inclusión de la ruta y el carácter separador de columna del fichero CSV asociado a un componente “FicheroEntrada” se hará desde el área de fichero del componente dentro del panel de gestión del componente y pulsando el botón “Aplicar”.				
Requisitos relacionados		RRTG-114, RRTG-115, RRTG-117, RRTG-125			

Tabla 175 - Requisito RRTG-138

Identificador	RRTG-139		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	La inclusión de la ruta del fichero CSV asociado a un componente “FicheroSalida” se hará desde el área de fichero del componente dentro del panel de gestión del componente y pulsando el botón “Aplicar”.				
Requisitos relacionados		RRTG-114, RRTG-115, RRTG-117, RRTG-128			

Tabla 176 - Requisito RRTG-139

Identificador	RRTG-140		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	La relación entre los campos de dos componentes de tipo “FicheroEntrada” se establecerá seleccionando el campo del primer componente, el campo del segundo componente e indicando si la unión es de tipo “Inner join” o “Left Join” y pulsando el botón con el símbolo de confirmación en el área de relaciones dentro del panel de gestión del componente de tipo “Mapeo”.				
Requisitos relacionados		RRTG-114, RRTG-115, RRTG-117, RRTG-126			

Tabla 177 - Requisito RRTG-140

Identificador	RRTG-141		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Descripción	El filtrado de un campo se hará seleccionando el nombre del campo, indicando si será igual o distinto su valor al indicado en el campo de filtro y pulsando el botón de confirmación dentro del panel de gestión del componente de tipo “Filtro”.				
Requisitos relacionados		RRTG-114, RRTG-115, RRTG-117, RRTG-127			

Tabla 178 - Requisito RRTG-141

Identificador	RRTG-142		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	El sistema permitirá subir una versión del trabajo al repositorio.				
Requisitos relacionados		RFTG-111, RRTG-112			

Tabla 179 - Requisito RRTG-142

Identificador	RRTG-143		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Descripción	El sistema incluirá un panel para gestionar las subidas a repositorio de un trabajo.				
Requisitos relacionados		RRTG-142			

Tabla 180 - Requisito RRTG-143

Identificador	RRTG-144		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	El panel para gestionar las subidas a repositorio de un trabajo incluirá un apartado para subir una versión del trabajo. Incluirá un campo para seleccionar la versión, un campo para incluir la descripción de la misma y un botón para subir la versión al repositorio.				
Requisitos relacionados		RRTG-142, RRTG-143			

Tabla 181 - Requisito RRTG-144

Identificador	RRTG-145		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja		
Descripción	El acceso al panel para gestionar subidas a repositorio de un trabajo se hará pulsando el botón “Repositorio” que está en la vista desde la que se elabora el proceso ETL dentro de un trabajo.				
Requisitos relacionados		RRTG-142, RRTG-143, RRTG-144			

Tabla 182 - Requisito RRTG-145

Identificador	RRTG-146		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Descripción	La subida de una versión de un trabajo al repositorio se hará seleccionando el número y la descripción de la versión y pulsando el botón “Subir” dentro del panel para gestionar las subidas a repositorio del trabajo. Requiere confirmación.				
Requisitos relacionados		RRTG-142, RRTG-143, RRTG-144, RRTG-145			

Tabla 183 - Requisito RRTG-146

Identificador	RRTG-147		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Descripción	El sistema permitirá sobrescribir la versión existente de un trabajo.				
Requisitos relacionados		RRTG-142, RRTG-143, RRTG-144, RRTG-145, RRTG-146			

Tabla 184 - Requisito RRTG-147

Identificador	RRTG-148		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Descripción	El sistema permitirá guardar los cambios de un trabajo como una copia del usuario de la aplicación.				
Requisitos relacionados		RFTG-111, RRTG-112			

Tabla 185 - Requisito RRTG-148

Identificador	RRTG-149		Categoría	Gestión de trabajos ETL	
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Riesgo	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Verificabilidad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media		<input type="checkbox"/> Baja	
Descripción	El almacenamiento de una copia del trabajo para el usuario se hará mediante el botón “Guardar” incluido en la vista desde la que se elabora el proceso ETL dentro de un trabajo. Mostrará un mensaje al usuario para que sepa que se ha efectuado la acción.				
Requisitos relacionados		RRTG-148			

Tabla 186 - Requisito RRTG-149

Anexo II. Control de versiones

Versión	Fecha de finalización	Detalle
0.1	13/06/2016	Elaboración: 1.Introducción
0.2	08/07/2016	Elaboración: 2.Estado de la cuestión
0.3	11/07/2016	Revisión del trabajo realizado hasta la fecha.
0.5	12/07/2016	Elaboración: 3.Gestión de proyecto software.
0.7	14/07/2016	Revisión del trabajo realizado hasta la fecha.
0.8	15/07/2016	Elaboración: 4.1.Definición del problema
0.9	28/07/2016	Elaboración: 4.2. Proceso de desarrollo, fase de Análisis: Requisitos y Casos de Uso.
1.0	31/07/2016	Revisión del trabajo realizado hasta la fecha.
1.3	01/08/2016	Elaboración: 4.2. Proceso de desarrollo, fase de Diseño: contexto del sistema, diseño del sistema y diseño detallado.
1.4	03/08/2016	Revisión del trabajo realizado hasta la fecha.
1.8	09/09/2016	Elaboración: 4.2. Proceso de desarrollo, fase de Implementación.
1.9	11/09/2016	Revisión del trabajo realizado hasta la fecha.
2.5	18/09/2016	Elaboración: 4.2. Proceso de desarrollo, fase de Pruebas. 5. Evaluación
2.6	19/09/2016	Revisión del trabajo realizado hasta la fecha.
2.7	20/09/2016	Elaboración: 6. Conclusión 7. Bibliografía
2.8	23/09/2016	Revisión exhaustiva de todos los capítulos del documento.

Tabla 187 - Control de versiones